
TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky a mezioborových studií

Studijní program: B2612 – Elektrotechnika a informatika

Studijní obor: 1802R022 – Informatika a logistika

Implementace mobilního IPv6

Implementation of mobile IPv6

Bakalářská práce

Autor: **Miroslav Laurin**

Vedoucí práce: Doc. RNDr. Pavel Satrapa, Ph.D.

V Liberci 16. 5. 2012

TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky a mezioborových studií
Akademický rok: 2011/2012

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Miroslav Laurin**
Osobní číslo: **M08000297**
Studijní program: **B2612 Elektrotechnika a informatika**
Studijní obor: **Informatika a logistika**
Název tématu: **Implementace mobilního IPv6**
Zadávající katedra: **Ústav nových technologií a aplikované informatiky**

Z á s a d y p r o v y p r a c o v á n í :

1. Seznamte se s problematikou podpory mobility v IPv6.
2. Proveďte rešerši dostupných programů.
3. Navrhněte infrastrukturu pro podporu mobility v domácí síti mobilních zařízení.
4. Návrh implementujte, ověřte a vyhodnoťte získané zkušenosti.

Prohlášení

Byl(a) jsem seznámen(a) s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu TUL.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Bakalářskou práci jsem vypracoval(a) samostatně s použitím uvedené literatury a na základě konzultací s vedoucím bakalářské práce a konzultantem.

Datum: 16. 5. 2012

Podpis: 

Poděkování

Na tomto místě bych rád poděkoval vedoucímu mé bakalářské práce Doc. RNDr. Pavlovi Satrapovi, Ph.D. za cenné rady, které mi při tvorbě této práce poskytl.

Abstrakt

Cílem této práce bylo prozkoumat úroveň podpory mobilního IPv6 na současných operačních systémech, porovnat dostupná řešení a na základě těchto informací ověřit funkčnost vybrané implementace.

Teoretická část práce pojednává o vzniku a vývoji mobility. Dále popisuje její základní princip a také se podrobněji zaměřuje na jednotlivé dílčí procesy mobility, jako je domácí registrace či optimalizace cesty.

V praktické části lze nalézt srovnání dostupných implementací mobility na běžně užívaných platformách. Největší prostor je pak věnován popisu konfigurace a výsledkům testování mobilní implementace na platformě Linux a MS Windows.

Klíčová slova: mobilita, mobilní IPv6, implementace, MIPL, Linux, MS Windows

Abstract

The aim of this study was to examine the level of support Mobile IPv6 on current operating systems, and compare available solutions based on this information to verify the functionality of the selected implementation.

The theoretical part deals with the origin and development of mobility. It also describes its the basic principle and is concerned with the mobility of individual sub-processes, such as home registration and route optimization.

Comparison of available implementations on commonly used platforms is described in the practical part. The largest area is devoted to describing the configuration and results of testing mobile implementation on Linux and MS Windows platforms.

Keywords: mobility, mobile IPv6, implementation, MIPL, Linux, MS Windows

Obsah

Prohlášení.....	3
Poděkování.....	4
Abstrakt.....	5
Obsah	6
Seznam obrázků	8
Seznam tabulek	8
Seznam zkratk	9
1 Úvod	10
2 Mobilita.....	11
2.1 Vznik a vývoj mobility v IPv6.....	11
2.2 Základní princip	12
2.3 Hlavička a rozšíření	15
2.4 Domácí agent	20
2.5 Optimalizace cesty	22
2.6 Reakce na změny	25
3 Mobilita v praxi.....	26
3.1 Dostupné implementace.....	26
3.1.1 MS Windows	26
3.1.2 Linux.....	27
3.1.3 Ostatní platformy	28
3.1.4 Výběr platformy pro testování	29
3.2 Návrh infrastruktury	30
3.2.1 Použitá technika.....	30
3.2.2 Zapojení a konfigurace	32
3.3 Ověření implementace	39
3.3.1 Linux.....	39
3.3.2 MS Windows	49
4 Závěr	51
Seznam použité literatury	53

Přílohy	54
A. Výpis programu radvdump	54
B. Výpis konfigurace síťových rozhraní	56
C. Konfigurace mobilní implementace – varianta IPsec	58
D. Výpisy směrovacích tabulek	61
E. Výpisy síťové komunikace z aplikace Ethereal (příloha na CD)	63

Seznam obrázků

Obr. 2.1: Navázání spojení s mobilním uzlem [3]	14
Obr. 2.2: Hlavička – Mobilita [3]	16
Obr. 2.3: Formát zprávy – aktualizace vazby [3]	17
Obr. 2.4: Formát zprávy – potvrzení vazby [3]	18
Obr. 2.5: Formát zprávy – žádost o adresy domácích agentů [3]	20
Obr. 2.6: Formát zprávy – odpověď na objevování domácího agenta [3]	21
Obr. 2.7: Formát zprávy – Zahájení testu domácí adresy [3]	23
Obr. 2.8: Formát zprávy – Test domácí adresy [3]	23
Obr. 2.9: Test zpětné směrovatelnosti [3]	24
Obr. 3.1: Schéma zapojení	32
Obr. 3.2: Výchozí stav v síti – test domácí registrace	40
Obr. 3.3: Znázornění výměny datagramů – domácí registrace mobilního uzlu	41
Obr. 3.4: Výchozí stav v síti – test optimalizace cesty	44
Obr. 3.5: Znázornění výměny datagramů – optimalizace cesty	45
Obr. 3.6: Výchozí stav v síti – návrat mobilního uzlu domů	47
Obr. 3.7: Znázornění výměny datagramů – návrat mobilního uzlu domů	48
Obr. 3.8: Znázornění výměny datagramů – optimalizace cesty (MS Windows)	50

Seznam tabulek

Tab. 2.1: Přehled prvků pro podporu mobility	15
Tab. 2.2: Typy zpráv [3]	17
Tab. 3.1: Porovnání platforem	28
Tab. 3.2: Seznam použitých počítačů	31
Tab. 3.3: Volby programu radvd	35
Tab. 3.4: Volby mobilní implementace (MIPL)	37
Tab. 3.5: Seznam voleb mobility systému Windows XP	39

Seznam zkratek

CN	Correspondent Node
ESP	Encapsulation Security Payload
ETSI	European Telecommunications Standards Institute
HA	Home Agent
ICMPv6	Internet Control Message Protocol version 6
IP	Internet Protocol
IPnG	IP Next Generation
IPsec	IP security
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
MIPL	Mobile IPv6 for Linux
MN	Mobile Node
RFC	Request for Comments
UTP	Unshielded Twisted Pair

1 Úvod

V této práci se věnuji podpoře mobility v internetovém protokolu verze 6. Díky ní lze bezpečně vytvořit stabilní a efektivní spojení s přenosným zařízením, přestože toto zařízení zrovna není ve své domácí síti na své domácí IP adrese, ale kdesi „na cestách“.

Informací o IPv6 mobilitě nenajdeme mnoho a kromě RFC (*Request for Comments*) dokumentů od samotných tvůrců, které jsou psány velmi odborně předpokládajíce jisté znalosti v oboru, najdeme jen několik autorů, kteří vydali vlastní publikaci obšírně popisující dané téma. Na domácí půdě existuje kupříkladu pouze jediná taková publikace.

První část této práce pojednává o samotném protokolu IPv6. Jelikož z něj mobilita vychází, považoval jsem za důležité zmínit jeho vývoj. Dále se věnuji již samotné mobilitě, jejímu vzniku, vývoji a také současnému stavu. V teoretické části vysvětluji základní princip mobility, ale pouštím se i do hloubky a detailněji popisuji funkci dílčích procesů mobilní komunikace.

V praktické části jsem zjišťoval, jak jsou na tom s podporou mobility běžně používané platformy, a provedl rešerši dostupných programů, které by případnou chybějící podporu doplnily. Na základě získaných informací jsem jako hlavní platformu pro návrh infrastruktury a ověření funkčnosti vybrané implementace zvolil Linux. Tento fakt byl sám o sobě pro mne výzvou, protože běžně Linux nepoužívám, a tak jsem se kromě samotné implementace mobility musel seznamovat i s ovládáním a nuancemi tohoto systému.

Přes mnohá úskalí se podařilo návrh implementovat, tudíž je závěr práce věnován výsledkům měření a získaným zkušenostem.

2 Mobilita

V dnešní době zažívají velký rozmach přenosná zařízení, která umožňují připojení k celosvětové síti počítačů – internetu. Nejedná se jen o notebooky coby typické představitele přenosného zařízení. Být „online“ vždy a všude je trendem moderní doby, a tak dnes možnost připojení k internetu najdeme i na každém modernějším telefonu, přehrávači hudby a zpravidla pak na tabletech, netbooku, apod. Díky bezdrátovým technologiím dnes máme k dispozici připojení k internetu na většině území naší země a nejinak je tomu ve vyspělých státech světa. Mobilní operátoři se chlubí sítěmi třetí generace, ve světě již existují sítě čtvrté generace s rychlostí přesahující 100Mbit/s. V restauracích a na dalších veřejných místech se často setkáváme Wi-Fi konektivitou.

Připojit se prakticky kdekoli do sítě internetu a komunikovat na svém přenosném zařízení s jiným zařízením v síti za předpokladu, že znám jeho IP adresu, není v zásadě problém. Potíže nastávají v momentě, kdy chce někdo navázat spojení se zařízením, které je v tu dobu na cestách, například ve vlaku jedoucím napříč Evropou, a nenachází se tedy na své obvyklé „domácí“ adrese. Navíc je zřejmé, že v takovém případě bude přenosné zařízení během cesty přecházet z jedné sítě do druhé, čímž se budou neustále měnit parametry sítě, zejména přidělená IP adresa. Taková situace nám velmi znesnadňuje možnost komunikace s tímto zařízením, o nepřerušovaném spojení nemluvě.

Elegantní řešení tohoto problému nabízí právě protokol IPv6 a podpora mobility v něm obsažená.

2.1 Vznik a vývoj mobility v IPv6

Internet Protocol verze 6 (*IPv6*) představuje další generaci protokolu IP a nástupce současně používaného Internet Protocol verze 4 (*IPv4*). Vývoj nového protokolu započal od roku 1991, tehdy pod označením IP Next Generation (*IPnG*).

Hlavním důvodem vzniku nového protokolu bylo zajištění dostatečně velkého adresního prostoru, protože již tehdy bylo jasné, že v rámci IPv4 nevystačí adresní prostor příliš dlouho. Kromě vyřešení problému s dostatkem adresního prostoru kladli tvůrci při návrhu důraz také na řešení problému adresace uzlů, dynamické konfigurace a podpory multimediálních aplikací, bezpečnostních procedur a mobilních zařízení.

V roce 1994 vzniklo RFC 1752 (*The Recommendation for the IP Next Generation Protocol*), které se stalo předlohou k finální verzi protokolu IPv6 s příslušným RFC 1883 (*Internet Protocol, Version 6(IPv6) Specification*). [1]

Podpora mobility není specialitou jen protokolu IPv6. Už verze 4 obsahovala jistou podporu mobility. Konkrétně o ní pojednává RFC 3344 (*IP Mobility Support for IPv4*). Prvotní diskuze o podpoře mobility v IPv4 pochází z roku 1993, kdy ještě prakticky žádné chytré telefony a tablety nebyly a za přenosné zařízení se považoval na dnešní dobu velký a těžký laptop, a tvůrci tak spíše předvíдали potřebu podpory mobilních zařízení někdy v budoucnu. Konečnou specifikaci mobility v IPv4 z roku 1996 obsahuje RFC 2002 (*IP Mobility Support*). [2]

O podpoře mobility v IPv6 se začalo mluvit v roce 1996, ale konečné specifikace se dočkala v dokumentu RFC 3775 (*Mobility Support in IPv6*) a s ním souvisejícím RFC 3776 (*Using IPsec to Protect Mobile IPv6 Signaling Between Mobile Nodes and Home Agents*) až v roce 2004. Prozatím konečná specifikace pochází z roku 2011 a odpovídá jí dokument RFC 6275 (*Mobility Support in IPv6*), který zohledňuje změny v IPv6, k nimž docházelo od roku 2004. Nicméně tyto změny mají spíše kosmetický charakter. Současný stav podpory mobility popisuje [3] slovy: „*Dodnes je bohužel stav podpory mobilního IPv6 dost neutěšený a zlepšuje se jen ppoommaalluu.*“

Zásadní vylepšení mobility v IPv6 oproti verzi 4 spočívá v optimalizaci cesty, čímž se značně zefektivnila cesta datagramu od odesílatele ke svému cíli.

2.2 Základní princip

Mobilita v IPv6 vychází z předpokladu, že mobilní zařízení je někde doma a má svoji domácí síť (*Home Network*), která mu přiřazuje stálou domácí adresu (*Home Address*). Mobilní zařízení se v terminologii IPv6 označuje jako mobilní uzel (*Mobile Node, MN*). Skrze domácí adresu je mobilní uzel vždy dostupný bez ohledu na to, zda se nachází, či nenachází v domácí síti. V momentě, kdy mobilní uzel opustí svou domácí síť a připojí se do jiné sítě (*Foreign Network, Visited Network*), obdrží další dočasnou adresu (*Care-of Address*), kterou mu přiřadí nová síť, do níž se připojil. [3]

Důležitou roli hraje domácí agent (*Home Agent, HA*), který zastupuje mobilní uzel, pokud není v domácí síti. Jedná se o směrovač v domácí síti, který zachytává datagramy posílané mobilnímu uzlu přes domácí adresu a přeposílá mu je tunelem. Mobilní uzel pozná, že datagramy k němu dorazily skrze tunel a že s ním chce někdo navázat spojení skrze jeho domácí adresu.

Korespondent (*Correspondent Node, CN*) – v terminologii IPv6 je takto označován partner, který chce komunikovat s mobilním uzlem – může díky této vlastnosti použít domácí adresu jako cílovou adresu mobilního uzlu, a to bez ohledu na jeho pozici či síť, do které je zrovna mobilní uzel připojen. Při komunikaci v opačném směru použije mobilní uzel jako adresu odesílatele svou domácí adresu a posílá datagramy zpět skrze tunel domácímu agentovi, který je následně přepośle korespondentovi.

Korespondentem může být každý uzel podporující IPv6 a může tak bez problému komunikovat s mobilním uzlem. Mobilita v IPv6 poskytuje zpětnou kompatibilitu všem IPv6 uzlům neohledně na to, zda podporují mobilitu. Nicméně pokud korespondent mobilitu nepodporuje, komunikace musí vždy probíhat tak, jak je popsána o odstavci výše, tzn. přes domácího agenta. To není příliš efektivní postup, obzvláště pokud se nachází mobilní uzel a korespondent nedaleko od sebe, v nejhorším případě ve stejné síti. [2]

Mechanismus řešící tento problém se nazývá optimalizace cesty (*Route Optimization*) a pro jeho využití ho musí podporovat obě strany, tedy i korespondent. Díky optimalizaci může mobilní uzel zasílat datagramy přímo korespondentovi za použití své dočasné adresy coby adresy odesílatele. Stejně tak korespondentovi je umožněna přímá komunikace s mobilním uzlem.

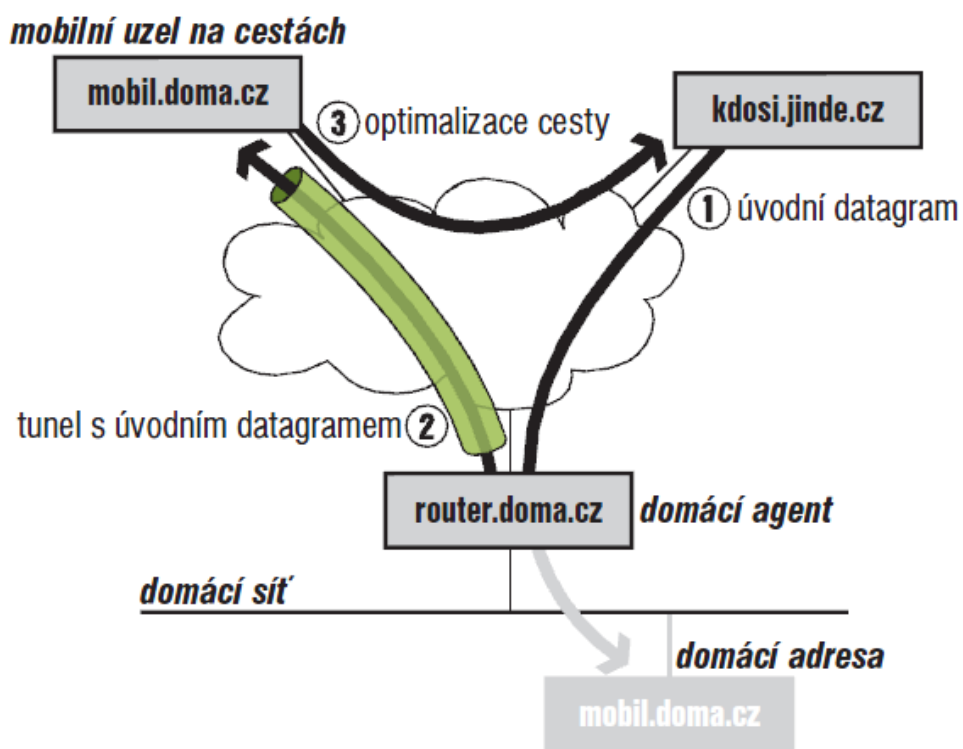
Optimalizaci zahajuje hned při přijetí prvního datagramu mobilní uzel, který odešle zprávu obsahující informaci o své aktuální dočasné adrese korespondentovi. Tato zpráva se nazývá aktualizace vazby (*Binding Update*). Tomu však musí předcházet ověřovací metoda – test zpětné směrovatelnosti (*Return Routability*) pro zjištění, zda je mobilní uzel opravdu tím, za co se vydává. V případě úspěchu přijme korespondent aktualizaci vazby a na oplátku pošle potvrzení vazby (*Binding Acknowledgment*). [3]

Další datagramy korespondent opatřuje rozšiřující hlavičkou směrování typu 2, kde uvede jako koncovou adresu domácí adresu mobilního uzlu, ale datagramy pošle na jeho dočasnou adresu.

Ve výsledku tedy mobilní uzel a korespondent komunikují přímo bez potřeby domácího agenta. Jak navázání spojení s optimalizací cesty vypadá, ilustruje Obr. 2..

Vztah mezi dočasnou a domácí adresou mobilního uzlu se nazývá vazba (*Binding*). Všechny uzly účastníci se mobilní komunikace mají za povinnost vést ve své datové struktuře tabulku zvanou cache vazeb (*Binding Cache*), která obsahuje informace o jednotlivých vazbách. [4]

Mobilita se umí vypořádat i se situací, kdy korespondent je sám o sobě mobilním uzlem. V tomto případě probíhá nepřerušovaná komunikace mezi oběma uzly bez ohledu na to, odkud jsou k internetu připojeni. [2]



Obr. 2.: Navázání spojení s mobilním uzlem [3]

2.3 Hlavička a rozšíření

RFC 3775 definuje novou rozšiřující hlavičku – mobilita (*Mobility Header*), která nese informace o vazbách a také testuje důvěryhodnost mobilního uzlu při optimalizaci cesty [3]. Dalším nově zavedeným prvkem je hlavička směrování typu 2 (*Type 2 Routing Header*). Tu využívá korespondent nebo domácí agent při komunikaci s mobilním uzlem. Tato hlavička stanovuje, že se má jako koncová adresa použít domácí adresa mobilního uzlu, ale datagramy se mají dopravit na jeho dočasnou adresu, kde dojde v hlavičce podle směrovacích pravidel k nahrazení dočasné adresy za domácí. Dále v rozšiřující hlavičce protokolu IPv6 – volby pro příjemce (*Destination Options*) přibyla volba – domácí adresa (*Home Address*) obsahující domácí adresu mobilního uzlu. Tuto volbu používá při odesílání datagramů mobilní uzel. Příjemce dostane datagram z dočasné adresy mobilního uzlu, ale v hlavičce ji nahradí domácí adresou, kterou si vyzvedne právě z volby domácí adresa. Tímto postupem je zajištěna transparentnost celého spojení z pohledu protokolů vyšších vrstev, které se tak o dočasně adrese nikdy nedovědí.

Dynamické objevování domácích agentů (*Dynamic Home Agent Address Discovery Request/Reply messages*) a objevování mobilních prefixů (*Mobile Prefix Solicitation/Advertisement messages*) pak zajišťují nově definované zprávy v protokolu ICMPv6 (*Internet Control Message Protocol version 6*). Přehled prvků zavedených pro podporu mobility naleznete v Tab. 2.. [3]

Tab. 2.: Přehled prvků pro podporu mobility

Hlavička – mobilita

- správa vazeb
- test domácí adresy
- test dočasné adresy

Hlavička směrování typu 2

- doručování dat mobilnímu uzlu

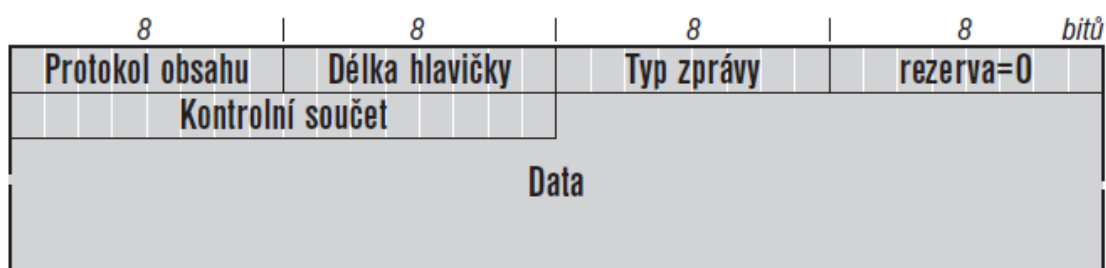
Volby pro příjemce

- domácí adresa

Protokol ICMPv6

- dynamické objevování domácího agenta
 - objevování mobilních prefixů
-

Jak již bylo naznačeno výše, hlavička mobilita se stará o správu vazeb. Strukturu hlavičky zobrazuje Obr. 2.. Její formát vychází ze standardního formátu rozšiřujících hlaviček. Položka protokol obsahu (*Payload Protocol*) nese informaci o následující hlavičce a významově se neliší od položky další hlavička (*Next Header*) používané ostatními rozšiřujícími hlavičkami. Nicméně v současné specifikaci není povoleno, aby za hlavičkou mobilita následovala další rozšiřující hlavička. Jinak řečeno, hlavička mobilita musí být vždy poslední hlavičkou v řetězci rozšiřujících hlaviček protokolu IPv6. Důvodem je zlepšení spolupráce mobility s IPsec (*IP security*) šifrováním. Délka hlavičky (*Header Length*) obsahuje délku hlavičky bez prvních 8 bajtů. [2]



Obr. 2.: Hlavička – Mobilita [3]

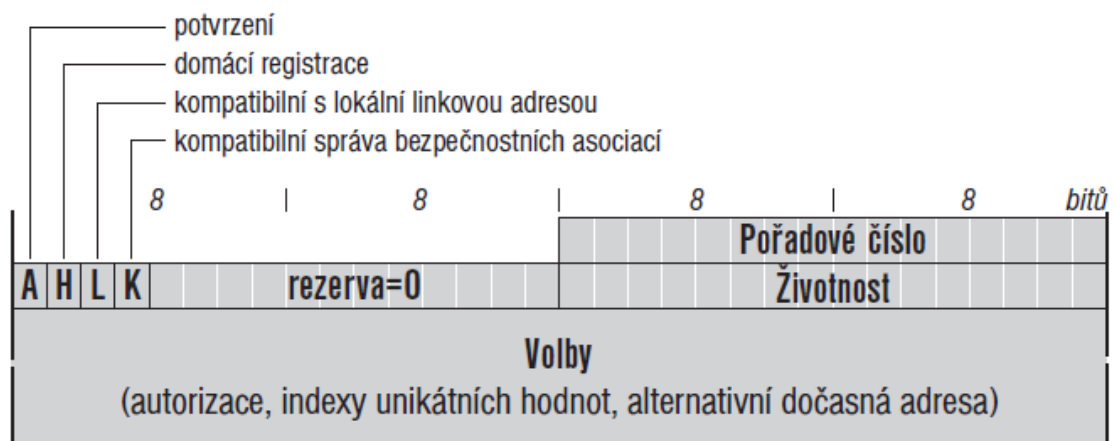
Za význačnou položku lze označit typ zprávy (*MH Type*), která stanovuje typ zprávy přenášené touto hlavičkou. Od ní se odvíjí i struktura přenášených dat (*Message Data*). Rozlišujeme osm základních typů zpráv, další byly definovány pozdějšími specifikacemi. Jejich soupis najdete v Tab. 2.. Struktura zpráv vymezuje zpočátku určitou pevnou část následovanou volbami. Ty mají také svůj vlastní formát společný pro všechny zprávy, ve kterých se mohou vyskytnout. Jaké volby budou použity, závisí na typu zprávy a také na aktuální situaci odesílatele. [3]

Položka rezerva (*Reserved*) je opravdu rezervou pro budoucí použití a v současnosti je její hodnota 0. Poslední položka hlavičky zahrnuje kontrolní součet (*Checksum*) přenášených dat.

Tab. 2.: Typy zpráv [3]

Typ	Význam
0	žádost o obnovení vazby (<i>binding refresh request</i>)
1	zahájení testu domácí adresy (<i>home test init</i>)
2	zahájení testu dočasné adresy (<i>care-of test init</i>)
3	test domácí adresy (<i>home test</i>)
4	test dočasné adresy (<i>care-of test</i>)
5	aktualizace vazby (<i>binding update</i>)
6	potvrzení vazby (<i>binding acknowledgement</i>)
7	chyba vazby (<i>binding error</i>)
8	rychlá aktualizace vazby (<i>fast binding update</i>)
9	rychlé potvrzení vazby (<i>fast binding acknowledgement</i>)
10	rychlé ohlášení souseda, zavrženo
11	experimentální hlavička
12	změna domácího agenta (<i>home agent switch</i>)
13	kontrola spojení (<i>heartbeat</i>)
14	zahájení předávky (<i>handover initiate</i>)
15	potvrzení předávky (<i>handover acknowledge</i>)
16	zrušení vazby (<i>binding revocation</i>)

Aktualizaci vazby používá mobilní uzel pro sdělení své aktuální adresy korespondentovi či domácímu agentovi. Formát zprávy znázorňuje Obr. 2..



Obr. 2.: Formát zprávy – aktualizace vazby [3]

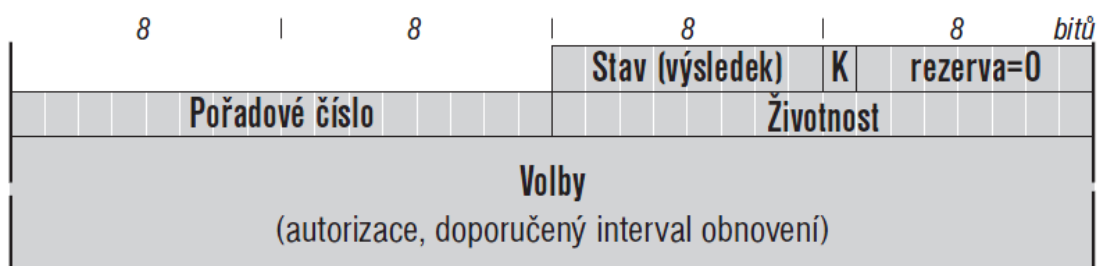
Jak píše [3], samotná aktualizace nenese informace o žádných adresách, protože ty se doplní ze zbytku datagramu. Z dočasné adresy mobilní uzel zprávu odesílá a domácí adresu přenáší ve volbě pro příjemce – domácí adresa, tudíž se příjemce obě adresy dozví.

Položka pořadové číslo (*Sequence #*) se s každou další aktualizací zvětšuje, aby se předešlo situaci, kdy se komunikační partner vrátí ke starší, již neplatné, dočasné adrese. Životnost (*Lifetime*) pak odpovídá době platnosti dočasné adresy v sekundách. Po vypršení platnosti se vazba považuje za neplatnou.

Pokud mobilní uzel opatří zprávu příznakem „A“, vyžaduje od příjemce potvrzení vazby. Naproti tomu příznakem „H“ požaduje po příjemci, aby se stal jeho domácím agentem. Nastavením příznaku „L“ dává mobilní uzel najevo, že má jeho domácí adresa stejný identifikátor rozhraní jako lokální linková adresa. Posledním příznakem „K“ sděluje mobilní uzel domácímu agentovi, že podporuje dynamickou správu bezpečnostních asociací pro IPsec odolný vůči přesunům uzlu v síti.

Zpráva může obsahovat volby autorizace (*Authorization Data*), index unikátních hodnot (*Nonce Indices*) a alternativní dočasná adresu (*Alternate Care-of Address*). Bližší popis je dostupný v RFC 6275.

Potvrzení vazby odesílá domácí agent nebo korespondent jako odpověď na aktualizaci vazby, pokud ta obsahovala příznak „A“ nebo „H“. Zpráva se odesílá i v případě zamítnutí aktualizace vazby (s informací o důvodu zamítnutí). Formát zprávy vidíme na Obr. 2..



Obr. 2.: Formát zprávy – potvrzení vazby [3]

Zdrojová adresa zprávy odpovídá adrese domácího agenta nebo korespondenta. Cílová adresa se rovná dočasné adrese mobilního uzlu s tím, že odesílatel musí použít hlavičku směrování typu 2 obsahující domácí adresu mobilního uzlu. [2]

Položka stav (*Status*) specifikuje výsledek zpracování aktualizace vazby. Pokud obsahuje hodnotu menší než 128, tak byla zpráva přijata. Hodnota 128 a vyšší znamená odmítnutí. Kompletní tabulku kódů lze nalézt v RFC 6275. Pořadové číslo (*Sequence #*) odpovídá poslednímu platnému pořadovému číslu v aktualizaci vazby. Životnost (*Lifetime*) značí dobu v sekundách, po kterou se příjemce zaručuje udržovat informace o vazbě. Příznakem „K“ sdělí domácí agent mobilnímu uzlu, zda bude využívat dynamickou správu bezpečnostních asociací.

U tohoto typu zprávy jsou dostupné volby autorizace (*Authorization Data*) a doporučený interval obnovení (*Binding Refresh Advice*).

Žádost o obnovení vazby se používá v případě, kdy partner mobilního uzlu potřebuje prodloužit platnost vazby, která by brzy skončila. Mobilní uzel, jenž tuto zprávu přijme, zareaguje odesláním zprávy – aktualizace vazby, čímž aktualizuje informace o vazbě na straně partnera. Také ale může poslat aktualizaci s nulovou životností, čímž vazbu zruší.

Zpráva neobsahuje žádné položky ani volby. Jako cílovou adresu použije partner domácí adresu mobilního uzlu. Partner nemůže při odesílání použít hlavičku směrování typu 2 ani volbu domácí adresa v hlavičce IPv6 – volby pro příjemce. Jelikož se nachází mobilní uzel mimo svou domácí síť, musí žádost o obnovení projít vždy cestou přes domácího agenta. [2]

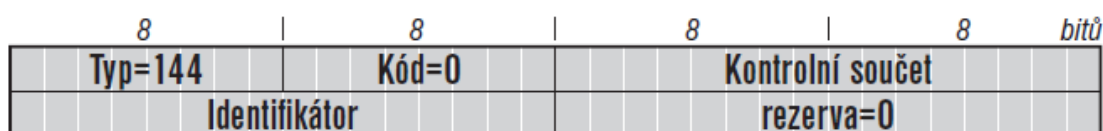
Zprávu **chyba vazby** používá korespondent k informování o vzniku chyby v procesu ustanovení mobilní komunikace. Zpráva obsahuje položku stav (*Status*), která informuje o důvodu vzniku této zprávy a položku domácí adresa (*Home Address*).

2.4 Domácí agent

Domácí agent zastupuje mobilní uzel, pokud se nenachází ve své domácí síti. Může ho vykonávat jeden či více směrovačů v domácí síti, jestliže podporují potřebnou konfiguraci a mají dostatek volných prostředků.

Jak píše [3], přiřazení agenta lze provést ručně statickou konfigurací v mobilním uzlu, což není příliš pohodlné při větším počtu zařízení, nehledě na to, že se situace v síti může časem měnit. Z tohoto důvodu byla vymyšlena metoda dynamické objevování domácího agenta (*Dynamic Home Agent Address Discovery*), která umožňuje mobilnímu uzlu najít vhodného agenta automaticky. Metoda využívá ICMPv6 zpráv.

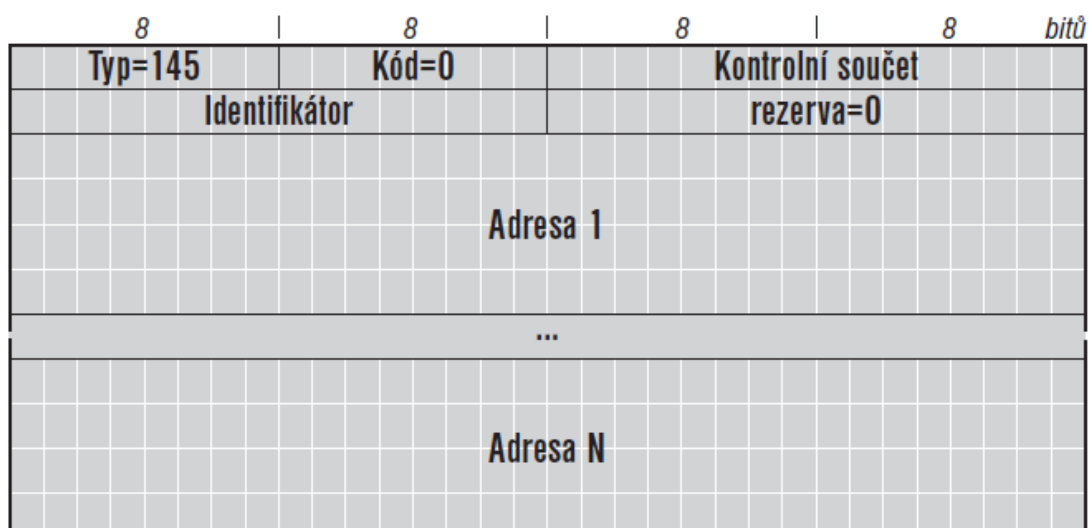
V prvním kroku odešle mobilní uzel ICMP zprávu (typ 144) – žádost o adresy domácích agentů (*ICMP Home Agent Address Discovery Request Message*) na výběrovou (*Anycast*) adresu domácích agentů v síti. Zpráva obsahuje jen položku identifikátor umožňující rozpoznat odpověď. Formát zprávy je patrný na Obr. 2..



Obr. 2.: Formát zprávy – žádost o adresy domácích agentů [3]

Každý směrovač v síti odesílá v rámci bezstavové konfigurace zprávu – ohlášení směrovače (*Router Advertisement*). Mobilita tuto zprávu rozšiřuje o několik prvků. Příznak „H“ znamená, že směrovač může v této síti sloužit jako domácí agent. Volbou informace o domácím agentovi (*Home Agent Information*) sděluje směrovač prioritu (*Home Agent Preference*) a životnost (*Home Agent Lifetime*) domácího agenta. Směrovače si navzájem informace z ohlášení druhých ukládají a udržují v seznamu domácích agentů (*Home Agents List*), takže o sobě mají jasný přehled.

Jelikož byla žádost o adresy domácích agentů odeslána na výběrovou (*Anycast*) adresu, chopí se odpovědi nejbližší směrovač (domácí agent) a odešle ICMP zprávu (typ 145) – odpověď na objevování adresy domácího agenta (*ICMP Home Agent Address Discovery Reply*). Formát zprávy ukazuje Obr. 2..



Obr. 2.: Formát zprávy – odpověď na objevování domácího agenta [3]

Odpověď nese seznam agentů v síti uspořádaných sestupně od toho s nejvyšší prioritou. Nicméně jak podotkl [3]: „*V reálném životě ale lze očekávat, že na lince bude jeden, nanejvýš několik málo domácích agentů.*“

Mobilní uzel si zpravidla vybere z odpovědi agenta s nejvyšší prioritou a tomu pošle aktualizaci vazby s příznakem „H“ stanovující, že se má právě on stát jeho domácím agentem. Tento agent provede kontrolu duplicity domácí adresy, zapíše si informace o mobilním uzlu do datových struktur (*Binding Cache*) a odešle tomuto uzlu potvrzení vazby. Pokud mobilní uzel nedostane potvrzení vazby, bude zkoušet odesílat aktualizaci znovu, dokud se tak nestane.

Jak píše ve své knize [2], proces ustanovení domácího agenta se nazývá domácí registrace (*Home Registration*).

Nyní už domácí agent pouze rozešle na skupinovou adresu pro všechny uzly (ff02::1) několik nevyžádaných ohlášení souseda (*Neighbor Solicitation*), v nichž sdělí IP adresu mobilního uzlu a svou linkovou adresu. Tím si zajistí předávání datagramů určených pro mobilní uzel.

Domácí agent také sleduje změny v síti. Pokud by tedy došlo například ke změně adres, může v potvrzení vazby požádat mobilní uzel, aby si vyhledal nové prefixy.

V tomto případě mobilní uzel reaguje zasláním ICMP zprávy – žádost o mobilní prefix (*Mobile Prefix Solicitation*), ve které žádá domácího agenta o poslání nových prefixů. Agent mu je obratem sdělí ve zprávě ohlášení mobilního prefixu (*Mobile Prefix Advertisement*).

Mobilní uzel by si měl hlídat životnost, po kterou je s domácím agentem dohodnuta spolupráce, a pokud v ní chce pokračovat, musí ještě před jejím vypršením požádat o obnovení vazby. V případě vypršení životnosti by mu totiž stávající agent vypověděl službu a on by si musel zopakovat kolečko objevování domácího agenta.

Pro zajištění bezpečnosti domácí registrace se využívá šifrovací mechanismus IPsec (*IP security*), konkrétně protokol ESP (*Encapsulation Security Payload*). Mezi mobilním uzlem a domácím agentem tak vznikne šifrováním chráněný tunel, tudíž se komunikace mezi nimi považuje za bezpečnou. [3]

2.5 Optimalizace cesty

Ustanovení přímé komunikace mezi mobilním uzlem „na cestách“ a korespondentem umožňuje proces zvaný optimalizace cesty (*Route Optimization*). [2]

Jak již bylo zmíněno v kapitole 2.2, bez optimalizace může probíhat komunikace mezi uzly pouze prostřednictvím domácího agenta, se kterým si mobilní uzel vytvořil vazbu. To z hlediska směrování není zdaleka ideální situace, speciálně pokud se nacházejí oba uzly topologicky blízko sebe, ba dokonce ve stejné síti.

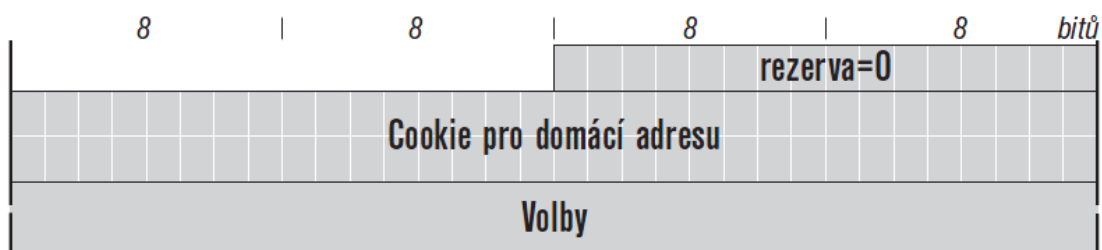
Proces optimalizace započne mobilní uzel v momentě, kdy mu dorazí první datagram od korespondenta. Jelikož se mobilní uzel nenachází v domácí síti, jedná se o tunelový datagram přeposlaný domácím agentem. Z toho mobilní uzel pozná, že se ho někdo snaží kontaktovat na domácí adrese, a pokusí se mu sdělit svou aktuální dočasnou adresu pomocí zprávy aktualizace vazby.

Tomu však musí předcházet ověřovací metoda zvaná zpětná směrovatelnost (*Return Routability*). Jinak by existovalo bezpečnostní riziko, kde by zrádný uzel X mohl předstírat, že je úplně jiný uzel Y mimo domov a zcizit tak jeho provoz.

Jak píše [3], mobilní uzel tedy musí prokázat, že je opravdu tím, za koho se vydává. V první fázi odešle zprávu – zahájení testu domácí adresy (*Home Test Init*) tunelem domácímu agentovi, který ji přepośle korespondentovi. Zároveň odešle i druhou

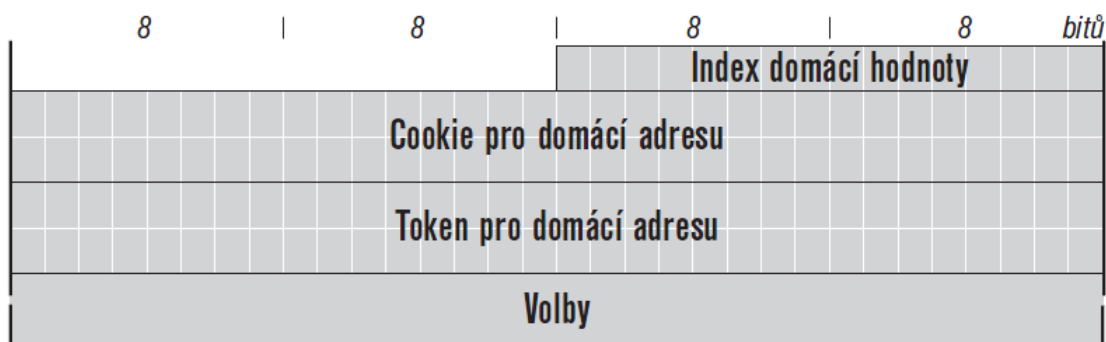
zprávu – zahájení testu dočasné adresy (*Care-of Test Init*) určenou přímo korespondentovi.

Obě zprávy obsahují náhodnou hodnotu *Cookie* (*Home Init Cookie*, *Care-of Init Cookie*), vygenerovanou mobilním uzlem. Formát zpráv je prakticky totožný, ukazuje ho Obr. 2..



Obr. 2.: Formát zprávy – Zahájení testu domácí adresy [3]

Korespondent zareaguje odesláním zpráv – test domácí adresy (*Home Test*) přes domácího agenta a test dočasné adresy (*Care-of Test*) přímo mobilnímu uzlu, ve kterých přepošle hodnoty *Cookie* vygenerované mobilním uzlem, čímž ho ujistí, že reaguje na jeho zahájení testu adres. Dále připojí hodnotu *Token*, kterou vygeneruje za pomoci několika unikátních hodnot, adres mobilního uzlu a hešovací funkce. Podrobnější popis generování *Token* naleznete v RFC 6275. Formát zprávy ukazuje Obr. 2..

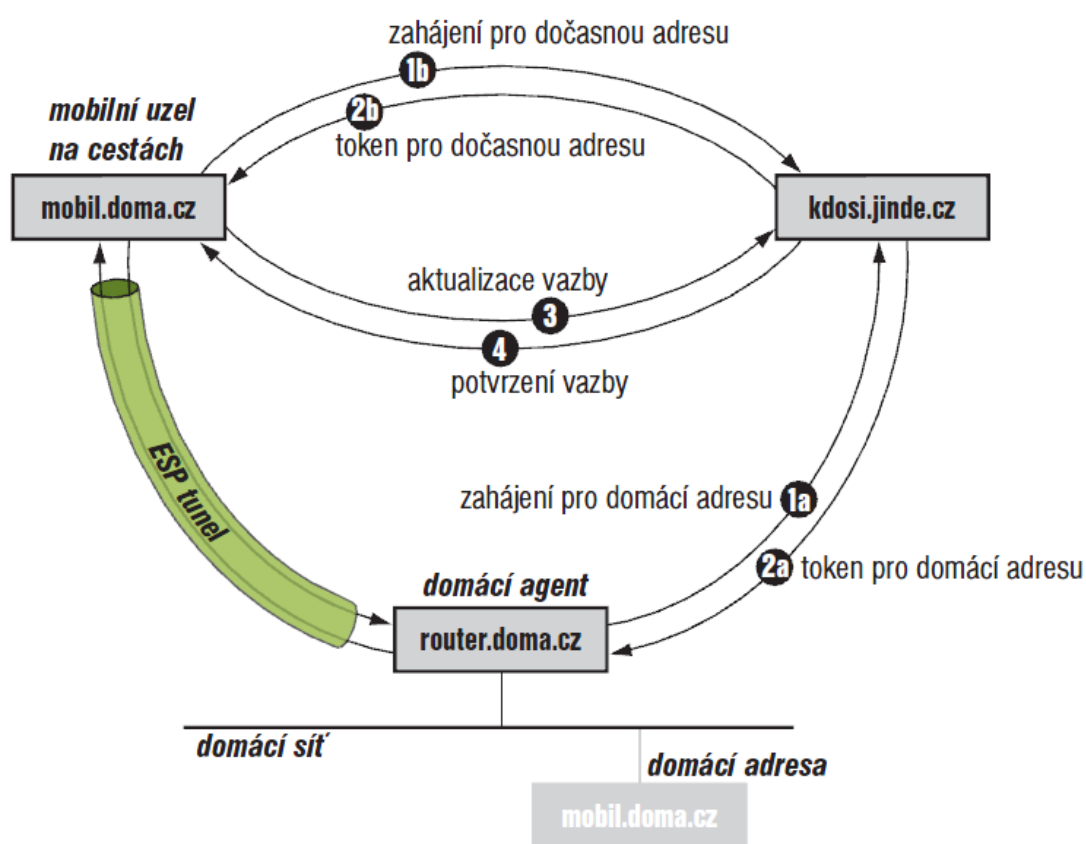


Obr. 2.: Formát zprávy – Test domácí adresy [3]

Mobilní uzel z obou odpovědí vyzvedne tokeny a daným způsobem z nich vypočítá autentizační hodnotu, kterou odešle v aktualizaci vazby ve volbě autorizační data přímo na adresu korespondenta.

Ten si podle daného postupu a informací z aktualizace také vypočítá autentizační hodnotu a porovná ji s hodnotou v aktualizaci od mobilního uzlu. Pokud se shodují, vytvoří si záznam v datové struktuře vazeb a konečně odešle potvrzení vazby. Tímto postupem tedy mobilní uzel potvrdil, že naslouchá na obou uvedených adresách, čímž úspěšně splnil test zpětné směrovatelnosti. Celý proces ukazuje Obr. 2.. [3]

Mobilní uzel je povinen vést seznam všech zaslaných aktualizací s jejich životností (*Binding Update List*), aby v případě změny dočasné adresy informoval korespondenty o nové adrese.



Obr. 2.: Test zpětné směrovatelnosti [3]

2.6 Reakce na změny

Na začátku kapitoly 2 jsem zmínil situaci, kdy mobilní uzel cestuje vlakem napříč Evropou, čímž poměrně často přestupuje mezi sítěmi jednoho i více operátorů. Velmi pravděpodobně mnohokrát změní svou dočasnou adresu, což by ale nemělo nijak ovlivnit ostatní partnery zapojené do mobilní komunikace.

V IPv6 sítích vysílají směrovače nepřetržitě v pravidelných intervalech zprávu – ohlášení směrovače (*Router Advertisement*) obsahující informace potřebné k automatické konfiguraci strojů v síti. Mobilní uzel si při vstupu do sítě vybere implicitní směrovač a z jeho ohlášení se dozví prefix sítě nutný pro tvorbu primární dočasné adresy, kterou poté nahlásí svému domácímu agentovi.

Jak uvádí [3], mobilní uzel i nadále sleduje ohlášení od implicitního směrovače, čímž zároveň kontroluje jeho dosažitelnost. Pokud po stanovenou dobu ohlášení nedostane, zkusí sám poslat zprávu – výzva sousedovi (*Neighbor Solicitation*). Pokud ani na tuto výzvu směrovač nezareaguje, mobilní uzel usoudí, že pravděpodobně došlo ke změně sítě a začne se shánět po ohlášení od jiných směrovačů, případně o něj sám požádá zprávu – výzva směrovači (*Router Solicitation*).

Na základě nových ohlášení si mobilní uzel zvolí nový implicitní směrovač a vytvoří novou dočasnou primární adresu, a to stejným postupem popsáním výše. Samozřejmě musí poslat novou adresu v aktualizaci vazby domácímu agentovi a všem korespondentům, jež si udržuje v seznamu aktualizací.

Specifickou událostí je návrat mobilního uzlu do své domácí sítě. V tomto případě musí zajistit vymazání cache vazeb u domácího agenta a všech korespondentů, čehož mobilní uzel docílí zasláním zprávy – aktualizace vazby s nulovou životností všem komunikačním partnerům. Zpráva pro domácího agenta musí navíc obsahovat příznak „H“ pro zrušení domácí registrace.

3 Mobilita v praxi

Po teoretickém úvodu, pojednávajícím o principech mobility, se v této části práce věnuji konkrétním implementacím mobilního IPv6.

3.1 Dostupné implementace

Podpora mobility v běžně používaných operačních systémech není nijak oslnivá a veřejně dostupných implementací třetích stran je poměrně málo. Může za to pravděpodobně pomalý nástup protokolu IPv6 a dlouhý proces vzniku specifikace mobility, která pochází teprve z roku 2004.

V hledání dostupných implementací jsem se soustředil zejména na dva u nás nejrozšířenější systémy – MS Windows a Linux. Nezanevřel jsem však úplně ani na ostatní platformy, kterým se věnuji v kapitole 3.1.3.

3.1.1 MS Windows

Společnost Microsoft se o podpoře mobility ve verzích systému Windows příliš nezmiňuje. Na oficiálních webových stránkách Microsoftu lze najít pouze dokument zvaný *Understanding Mobile IPv6*¹, kde se kromě obecného popisu mobility, převzatého z RFC 3775, nedozvíme vůbec nic o možnostech konfigurace ve Windows.

Z různých neoficiálních zdrojů na internetu vyšlo najevo, že jistou omezenou podporu mobility obsahují systémy Windows XP/2003. Konkrétně se jedná o podporu korespondenta (*Correspondent Node*). Roli domácího agenta či mobilního uzlu platforma MS Windows neumožňuje. Nicméně podpora korespondenta nám aspoň dává naději využít proces optimalizace cesty k ustanovení přímé komunikace s jiným mobilním uzlem.

V novějších verzích Windows (Vista/7/2008) nenajdeme podporu mobility paradoxně vůbec, vývojáři ji z nějakého důvodu zcela odstranili. Nezbyvá než doufat, že další generace Windows na tom budou o něco lépe.

¹ <http://www.microsoft.com/en-us/download/details.aspx?id=10905>

3.1.2 Linux

V současnosti žádná běžně používaná distribuce Linuxu neobsahuje v základním sestavení podporu mobility. V tomto ohledu jsou si oba systémy rovny.

Situaci Linuxu vylepšuje fakt, že pro něj existuje několik projektů, jež podporu mobility do systému přidávají. Všechny projekty vyžadují úpravu jádra (*kernel*) Linuxu a jeho následnou rekompilaci.

Projekt Lancaster University v Anglii² se již považuje za hodně zastaralý a uvádím ho zde spíš jen pro úplnost. Vývoj se zastavil v roce 1998 a jeho webové stránky už nejsou funkční. Poslední podporovaná verze jádra je 2.1.90.

Za dalším počinem stojí vývojáři z Helsinky University of Technology³ (dnes Aalto University School of Science and Technology). Projekt nese název MIPL Mobile IPv6 for Linux. Bohužel ani ten již není moc aktuální. Vývoj skončil v roce 2006 a webové stránky projektu už také nejsou dostupné. Poslední verze projektu 2.0.2 byla určena pro jádro ve verzi 2.6.16.

Pravděpodobně nejaktuálnějším projektem (poslední aktualizace proběhla v roce 2011) je UMIP.org⁴, který navazuje na výše zmíněný projekt MIPL Mobile IPv6 for Linux. Oproti němu přidává zejména podporu pro mobilní síť NEMO a opravuje zjištěné nedostatky.

Pod záštitou ETSI⁵ (*European Telecommunications Standards Institute*) vznikl ještě jiný zajímavý projekt. Autoři zde využili výše zmíněnou implementaci MIPL Mobile IPv6 for Linux ve verzi 2.0.1, dále přidali program *radvd* 0.9.1 (*Linux IPv6 Router Advertisement Daemon*) sloužící pro vysílání zpráv ohlášení směrovače a spolu s několika dalšími úpravami vytvořili vlastní live distribuci s podporou mobility – Knoppix MIPL založenou na standardní distribuci Knoppix (v4.0.2-2005-09-23).

Stránky projektu jsou aktuálně funkční a můžeme na nich nalézt popis projektu a odkaz ke stažení obrazu CD s modifikovanou distribucí včetně podrobné dokumentace popisující konfiguraci mobilní implementace.

² <http://www.cs-ipv6.lancs.ac.uk/MobileIP>

³ <http://www.mobile-ipv6.org>

⁴ <http://www.umip.org/>

⁵ <http://www.ipt.etsi.org/knoppix.htm>

3.1.3 Ostatní platformy

BSD patří mezi systémy odvozené z Unixu. Jeho zastoupení na trhu je velmi malé a celkově se o něm příliš nemluví.

Pro systémy BSD se do roku 2006 vyvíjel japonský projekt KAME⁶, který řešil podporu IPv6 včetně mobility. V novějších variantách BSD (FreeBSD, OpenBSD, NetBSD, BSD/OS) se stal jeho kód standardní součástí jádra.

Mezi největší výrobce síťových zařízení patří společnost **Cisco**, jejíž nejvyšší řada produktů s operačním systémem IOS (verze 12.3 a výše) zahrnuje implementaci mobilního IPv6.

Směrovače Cisco lze nastavit jako domácího agenta. Roli mobilního uzlu či korespondenta nepodporují. Výhodu toho řešení vidím ve spojení funkce směrovače s rolí domácího agenta do jednoho zařízení. Na druhou stranu tato výhoda něco stojí, směrovače řady IOS patří do vyšší cenové kategorie.

Porovnání popisovaných implementací shrnuje Tab. 3..

Tab. 3.: Porovnání platforem

Platforma	MS Windows	Linux	BSD	Cisco
Systém	2003 XP(≥SP1)	kernel (≤v2.6.16) Knoppix MIPL ⁷	FreeBSD (≥v4.0) OpenBSD (≥v2.7) NetBSD (≥v1.5) BSD/OS (≥v4.2)	IOS (≥v12.3)
Nativní podpora mobility	Ano	Ne Ano ⁷	Ano	Ano
Implementace	Microsoft	MIPL Mobile IPv6	KAME	Cisco
Domácí agent	Ne	Ano	Ano	Ano
Mobilní uzel	Ne	Ano	Ano	Ne
Korespondent	Ano	Ano	Ano	Ne

⁶ <http://www.kame.net>

⁷ Upravená live distribuce Knoppix (projekt ETSI)

3.1.4 Výběr platformy pro testování

Při výběru vhodného kandidáta pro praktické vyzkoušení mobility jsem vycházel z výše uvedených dostupných řešení, jejichž seznam jsem dále zúžil podle následujících hledisek:

- rozšířenost platformy
- požadavky na provoz
- zkušenosti s platformou

Do testů jsem zařadil platformu **MS Windows**, konkrétně jsem ověřoval podporu korespondenta v systému Windows XP SP3, kterou údajně obsahuje.

Rodina systému Windows drží přibližně 90% podílu na trhu, tudíž nelze tuto platformu jen tak ignorovat. Na provoz systému není zapotřebí žádný speciální hardware, postačí běžný průměrně výkonný počítač s architekturou x86. Počítače se systémy Microsoftu používám několik let na administrátorské úrovni, tudíž své zkušenosti považuji za dostačující.

Za hlavní testovací platformu jsem zvolil **Linux**. Využil jsem upravené live distribuce Knoppix (projekt ETSI), která slibuje podporu všech režimů mobility – domácího agenta, mobilní uzel i korespondenta.

Linux zaujímá na trhu minoritní postavení v řádu jednotek procent, přesto se však jedná o třetí nejpoužívanější desktopový systém. Požadavky na systém se nijak neliší od platformy Windows, na provoz a testování postačí běžný PC. Linux nepatří mezi mnou běžně používané systémy, avšak jisté základní znalosti o něm jsem získal během studia na vysoké škole.

Platformu **BSD** jsem z výběru vyloučil. Jednak s ohledem na mizivé zastoupení na trhu a také z důvodu nedostatku mých zkušeností s tímto systémem.

Implementaci postavenou na platformě **Cisco** jsem také musel vyřadit z důvodu absence směrovače s operačním systémem IOS a možnosti jeho pořízení vzhledem k vysokým pořizovacím nákladům.

3.2 Návrh infrastruktury

Hlavní testovací platforma Linux vyžadovala pro ověření všech režimů mobility použití tří počítačů, které zastupovaly jednotlivé role mobilní komunikace, tedy domácího agenta, mobilní uzel a korespondenta. MS Windows podporuje pouze roli korespondenta, tudíž v tomto případě stačil jeden počítač postavený na této platformě. Zbylé role musela opět zajistit platforma Linux.

Jak jsem již uvedl v předešlé kapitole 3.1.4, žádný ze systémů si neklade přílišné požadavky na hardware a spokojí se s běžným průměrně výkonným PC. Všechny počítače musely obsahovat síťovou kartu, avšak její přítomnost považuji v dnešní době již za standard.

Návrh síťové infrastruktury spočíval ve vytvoření dvou IPv6 sítí s odlišným adresovým prostorem, jež by představovaly ekvivalent domácí (*Home Network*) a navštívené sítě (*Visited Network*). Dále bylo nutno zajistit správné směrování mezi sítěmi a bezstavovou automatickou konfiguraci klientů.

K tomuto účelu jsem využil řešení od autorů stojících za upravenou distribucí Knoppix, kteří zakomponovali do distribuce program radvd sloužící k odesílání zpráv ohlášení směrovače a umožnili tak postavit počítač do role směrovače. Jelikož jsem ale neměl pro tuto roli k dispozici další vhodný počítač, pátral jsem na základě dostupné dokumentace [5] po možnosti sjednocení role směrovače a domácího agenta do jednoho zařízení. Spojením konfigurace obou rolí se tato možnost ukázala jako proveditelná.

3.2.1 Použitá technika

Testování zvolených implementací probíhalo v domácích podmínkách za použití techniky, kterou se podařilo v rámci možností a omezených zdrojů sehnat. Celkově si měření vyžádalo použití tří počítačů a dvou prepínačů. Propojení všech zařízení bylo realizováno pomocí UTP kabelů (Cat 5E).

Pro snadnější orientaci v textu jsem počítače očísloval a stejné číslování budu používat i v dalších kapitolách. Soupis počítačů se specifikacemi rovněž shrnuje Tab. 3..

Knoppix patří mezi takzvané „Live CD“ distribuce, což znamená, že nevyžaduje instalaci na pevný disk. Systém jsem proto na počítače neinstaloval, nýbrž rovnou inicioval z CD.

Systém Windows XP SP3 byl instalován na **počítač č.3**, který tím pádem poskytoval bázi pro obě platformy.

Počítač č.1 musel být pro svou roli směrovače a domácího agenta dovybaven druhou síťovou kartou, aby mohl poskytnout dvě síťová rozhraní potřebná pro vytvoření domácí a navštívené sítě.

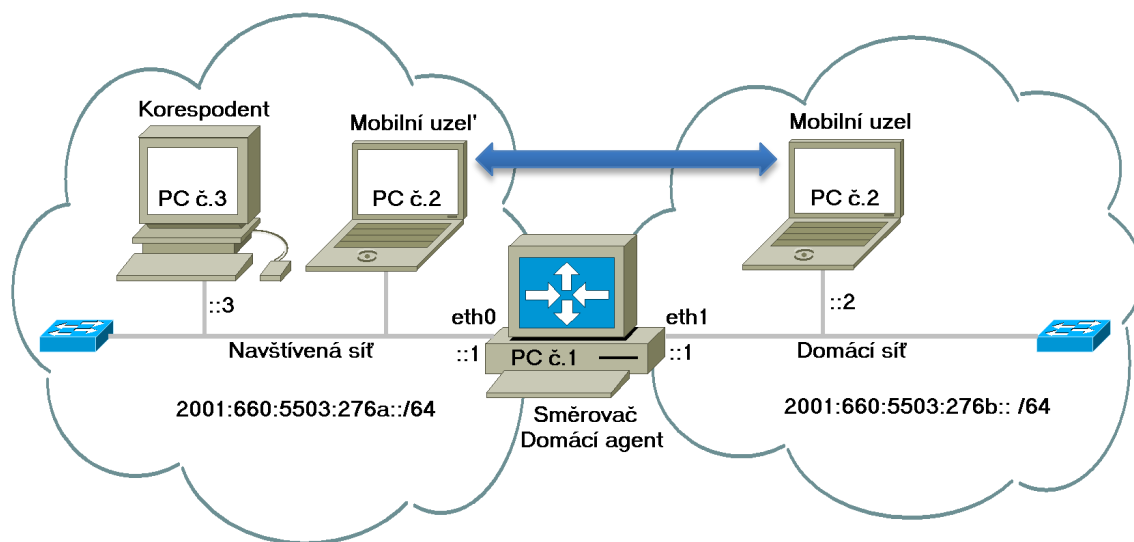
Tab. 3.: Seznam použitých počítačů

	Počítač č.1	Počítač č.2	Počítač č.3
Typ PC	Stolní PC	Notebook	Stolní PC
Procesor	Intel P4 1,4GHz	Intel Core Duo 2GHz	Intel Celeron 2,8GHz
Paměť RAM	768 MB	4096 MB	512 MB
Síť	Intel PRO/100VE Realtek 8139/810x	NetXtreme Gigabit PCIe	Realtek 8139 Family
OS	Knoppix MIPL	Knoppix MIPL	Knoppix MIPL MS Windows XP SP3
Implementace	MIPL Mobile IPv6	MIPL Mobile IPv6	MIPL Mobile IPv6 Microsoft
Role	Směrovač Domácí agent	Mobilní uzel	Korespondent

Součástí infrastruktury domácí sítě tvořil přepínač SMC FS5 (10/100MB). V případě navštívené sítě se jednalo o přepínač D-Link DSG-1008D (10/100/1000MB).

3.2.2 Zapojení a konfigurace

Zapojení počítačů v síti se z hlediska platformy nijak nelišilo. Mobilní uzel se stal jediným pohyblivým prvkem a během testování byl dle aktuální fáze testu přepojován z domácí sítě do navštívené a naopak. Celé zapojení nejlépe vystihuje schéma patrné na Obr. 3..



Obr. 3.: Schéma zapojení

Linux

Při konfiguraci počítačů postavené na platformě Linux jsem vycházel z dokumentace od autorů projektu Knoppix MIPL [5].

V prvním kroku bylo zapotřebí nastavit síťová rozhraní na všech počítačích tak, aby vznikla plně fungující IPv6 síť. Při správné konfiguraci musí být všechny uzly navzájem dosažitelné, což lze ověřit například příkazem `ping6`. „Viditelnost“ uzlů v síti je klíčovým předpokladem pro pokračování v testu. Pokud by kupříkladu mezi mobilním uzlem a korespondentem neprošlo ověření dosažitelnosti příkazem `ping6`, těžko by později mezi nimi mohlo dojít k optimalizaci cesty.

Dále již uvádím konkrétní konfiguraci síťových rozhraní jednotlivých počítačů. Nastavení bylo provedeno prostřednictvím linuxové konzole, která patří mezi standardní součást systému.

Počítač č.1 (směrovač / domácí agent):

- ✓ aktivace síťových rozhraní (eth0 → navštívená síť, eth1 → domácí síť)

```
ifconfig eth0 up  
ifconfig eth1 up
```
- ✓ nastavení zvolených IPv6 adres oběma rozhraním

```
ifconfig eth0 inet6 add 2001:660:5503:276a::1/64  
ifconfig eth1 inet6 add 2001:660:5503:276b::1/64
```
- ✓ nastavení předávání síťového provozu mezi rozhraními => funkce směrovače

```
echo "1" > /proc/sys/net/ipv6/conf/all/forwarding  
echo "0" > /proc/sys/net/ipv6/conf/all/accept_ra  
echo "0" > /proc/sys/net/ipv6/conf/all/accept_redirects  
echo "0" > /proc/sys/net/ipv6/conf/all/autoconf
```

Počítač č.2 (mobilní uzel):

- ✓ aktivace síťového rozhraní

```
ifconfig eth0 up
```
- ✓ povolení příjmu zpráv ohlášení směrovače a požadavků na přesměrování
nastavení automatické konfigurace na základě informací z ohlášení směrovače

```
echo "0" > /proc/sys/net/ipv6/conf/eth0/forwarding  
echo "1" > /proc/sys/net/ipv6/conf/eth0/accept_ra  
echo "1" > /proc/sys/net/ipv6/conf/eth0/accept_redirects  
echo "1" > /proc/sys/net/ipv6/conf/eth0/autoconf
```

Počítač č.3 (korespondent):

- ✓ aktivace síťového rozhraní

```
ifconfig eth0 up
```
- ✓ nastavení zvolené IPv6 adresy

```
ifconfig eth0 inet6 add 2001:660:5503:276a::3/64
```
- ✓ povolení příjmu zpráv ohlášení směrovače a požadavků na přesměrování
nastavení automatické konfigurace na základě informací z ohlášení směrovače

```
echo "0" > /proc/sys/net/ipv6/conf/eth0/forwarding  
echo "1" > /proc/sys/net/ipv6/conf/eth0/accept_ra  
echo "1" > /proc/sys/net/ipv6/conf/eth0/accept_redirects  
echo "1" > /proc/sys/net/ipv6/conf/eth0/autoconf
```

Další krok konfigurace spočíval v nastavení počítače č.1 do aktivní role směrovače a domácího agenta. K tomuto účelu jsem využil program radvd sloužící k odesílání zpráv ohlášení směrovače. Soubor s konfigurací definující funkci programu musí být uložen v následujícím umístění – */usr/local/etc/radvd.conf*.

Níže uvádím mnou použitou konfiguraci programu radvd pro jednotlivá rozhraní (eth0 → navštívená síť, eth1 → domácí síť). Význam použitých voleb pak shrnuje Tab. 3..

Nastavení programu radvd (počítač č.1):

```
interface eth0
{
    AdvSendAdvert on;
    AdvIntervalOpt on;

    MinRtrAdvInterval 1;
    MaxRtrAdvInterval 3;
    AdvHomeAgentFlag off;

    prefix 2001:660:5503:276a::/64
    {
        AdvOnLink on;
        AdvAutonomous on;
        AdvRouterAddr on;
    };
};

interface eth1
{
    AdvSendAdvert on;
    AdvIntervalOpt on;

    MinRtrAdvInterval 1;
    MaxRtrAdvInterval 3;
    AdvHomeAgentFlag on;
    HomeAgentLifetime 10000;
    HomeAgentPreference 20;
    AdvHomeAgentInfo on;

    prefix 2001:660:5503:276b::/64
    {
        AdvOnLink on;
        AdvAutonomous on;
        AdvRouterAddr on;
        AdvPreferredLifetime 10000;
        AdvValidLifetime 12000;
    };
};
```

Tab. 3.: Volby programu radvd

Volby rozhraní	Význam
AdvSendAdvert	odesílání zpráv ohlášení směrovače (on,off)
AdvIntervalOpt	zahrnutí volby – interval ohlášení do ohlášení směr. (on, off)
MinRtrAdvInterval	nejkratší povolená doba mezi odesíláním ohlášení (s)
MaxRtrAdvInterval	nejdelší povolená doba mezi odesíláním ohlášení (s)
AdvHomeAgentFlag	propagace schopnosti stát se domácím agentem (on, off)
HomeAgentLifetime	doba přístupnosti směrovače vykonávat domácího agenta (s)
HomeAgentPreference	preference domácího agenta (číslo, integer)
AdvHomeAgentInfo	zahrnutí informace o domácím agentovi do ohlášení (on, off)

Volby prefixu	Význam
AdvOnLink	prefix lze použít pro identifikaci na lince (on, off)
AdvAutonomous	prefix lze použít pro bezstavovou konfiguraci (on, off)
AdvRouterAddr	zasílání celé adresy rozhraní, ne jen prefix (on, off)
AdvPreferredLifetime	platnost adresy vytvořené pomocí bezstavové konfigurace (s)
AdvValidLifetime	platnost prefixu uděleného za účelem identifikace na lince (s)

Konfigurace rozhraní domácí sítě (eth1) obsahuje rovněž volby související s funkcí domácího agenta. Tyto volby jsou v Tab. 3. zvýrazněny tučným fontem.

Program radvd se aktivuje příkazem `/etc/init.d/radvd start`, čímž dojde k zahájení odesílání zpráv ohlášení směrovače na základě uvedené konfigurace. Zda program pracuje správně, lze ověřit příkazem `radvdump`. Po jeho zadání by se měla v konzoli periodicky vypisovat zpráva s konfigurací všech rozhraní, což se v rámci mého testování potvrdilo (viz. příloha A). Mobilní uzel a korespondent zareagují na spuštění programu radvd zahájením bezstavové konfigurace síťových rozhraní na základě informací získaného z ohlášení směrovače, čímž získají další IPv6 adresu s prefixem odpovídajícím rozhraní směrovače.

Finálním krokem konfigurace bylo nastavení mobilní implementace (MIPL) na každém počítači dle jeho role v mobilitě. Soubor s konfigurací je vždy nutno uložit do toho umístění – `/usr/local/etc/mip6d.conf`. Dále v textu předkládám mnou použité nastavení mobilní implementace. Význam použitých voleb lze najít v Tab. 3..

Počítač č.1 (domácí agent):

```
NodeConfig HA;  
DebugLevel 10;  
DoRouteOptimizationCN enabled;  
  
Interface "eth1";  
UseMnHaIPsec disabled;
```

Počítač č.2 (mobilní uzel):

```
NodeConfig MN;  
DebugLevel 10;  
DoRouteOptimizationCN enabled;  
  
Interface "eth0";  
UseMnHaIPsec disabled;  
  
DoRouteOptimizationMN enabled;  
UseCnBuAck enabled;  
  
MnHomeLink "eth0" {  
    HomeAgentAddress 2001:660:5503:276b::1;  
    HomeAddress 2001:660:5503:276b::2/64;  
}
```

Počítač č.3 (korespondent):

```
NodeConfig CN;  
DebugLevel 10;  
DoRouteOptimizationCN enabled;
```

Mobilní implementace (*MIPL*) se na všech počítačích aktivuje jednotným příkazem `/etc/init.d/mip6d start`. Tímto krokem byla také završena celá konfigurace na platformě Linux. Prvotní ověření korektního spuštění mobilní implementace lze provést na počítači č.2, představujícím mobilní uzel, příkazem `ifconfig`. Zadáním toho příkazu se v konzoli vypíše seznam rozhraní, mezi nimiž by mělo nově figurovat automaticky vytvořené virtuální rozhraní `ip6tnl1`, jak tomu bylo v případě mého testování:

```
ip6tnl1  Link encap:UNSPEC  HWaddr 20-01-06-60-55-03-27-6B-00-  
00-00-00-00-00-00-00  
        inet6 addr: fe80::217:8ff:fe30:2d4/64 Scope:Link  
        UP POINTOPOINT RUNNING NOARP  MTU:1460  Metric:1  
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
        collisions:0 txqueuelen:0  
        RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
```

Kompletní výpis příkazu `ifconfig` z konzole, zobrazující informace o výsledné konfiguraci všech rozhraní v systému, lze pro všechny počítače nalézt v příloze B.

Tab. 3.: Volby mobilní implementace (MIPL)

Společné volby pro všechny uzly	Význam
NodeConfig	určení role mobilní implementace (HA, MN, CN) ⁸
DebugLevel	úroveň ladění mobilní implementace (číslo)
DoRouteOptimizationCN	podpora optimalizace cesty z role CN (boolean)
Domácího agent a mobilní uzel	Význam
Interface	výběr síťového rozhraní (název rozhraní)
UseMnHalPsec	použití šifrování IPsec mezi HA-MN (boolean)
Mobilní uzel	Význam
DoRouteOptimizationMN	podpora optimalizace cesty z role MN (boolean)
UseCnBuAck	použití příznaku „A“ v aktualizaci vazby (boolean)
MnHomeLink	výběr síťového rozhraní (název rozhraní)
HomeAgentAddress	adresa domácího agenta (IPv6 adresa)
HomeAddress	domácí adresa MN (IPv6 adresa)

Za běžných podmínek by se měla veškerá komunikace mezi domácím agentem a mobilním uzlem chránit šifrovacím mechanismem IPsec, konkrétně hlavičkou ESP. Testování jsem uskutečnil ve variantách s šifrováním IPsec i bez šifrování. Nicméně z hlediska prezentace výsledků se jevila jako vhodnější varianta bez šifrování. V opačném případě by totiž hlavička ESP skryla kupříkladu podstatné informace pro ověření principu domácí registrace mobilního uzlu u domácího agenta. Konfigurace mobilní implementace na straně 35 vystihuje variantu bez šifrování. Variantu konfigurace s šifrováním lze najít v příloze C.

MS Windows

Testování na platformě MS Windows spočívalo v ověření podpory role korespondenta v systému Windows XP. Ostatní role mobility systém nepodporuje, a proto domácího agenta a mobilní uzel musela opět zastat platforma Linux. Konfigurace počítačů č.1 a č.2 se proto nijak nezměnila a nově bylo třeba nastavit pouze počítač.3 odpovídající roli korespondenta.

⁸ HA – domácí agent; MN – mobilní uzel; CN – korespondent

V systému bylo nutno ze všeho nejdříve nainstalovat podporu protokolu IPv6. Poté již následovala automatická konfigurace síťového rozhraní. Nakonec se musela ještě povolit role korespondenta mobilního IPv6, která je v systému v základu deaktivována. Konkrétní příkazy, které jsem použil při konfiguraci, prezentuji níže. Dostupné volby mobility v systému Windows XP vystihuje Tab. 3.. K zadávání příkazů se v systémech MS Windows používá příkazový řádek (*cmd*).

Počítač č.3 (korespondent):

- ✓ instalace podpory protokolu IPv6

```
ipv6 install
```

- ✓ vypnutí generování dočasných adres

```
netsh interface ipv6 set privacy disabled
```

- ✓ přidání vlastní IPv6 adresy

```
netsh interface ipv6 add address "sit" 2001:660:5503:276a::3
```

- ✓ povolení role korespondenta

```
netsh interface ipv6 set mobility correspondentnode=enabled
```

Po tomto nastavení by měl být systém připraven na roli korespondenta. Ověření správné konfigurace protokolu IP a mobility lze provést následujícími příkazy (za příkazem vždy uvádím konkrétní výstup z mého testování):

- ✓ výpis konfigurace protokolu IP

```
ipconfig
```

Adaptér sítě Ethernet sit:

```
Přípona DNS podle připojení . . . :  
P adresa automatické konfigurace : 169.254.132.137  
Maska podsítě . . . . . : 255.255.0.0  
Adresa IP . . . . . : 2011:660:5503:276a::3  
Adresa IP . . . . . : fe80:2e0:4cff:fedb:1a16%4  
Adresa IP . . . . . : fe80:20e:2eff:fe6c:c0b5%4
```

- ✓ výpis konfigurace mobility

```
netsh interface ipv6 show mobility
```

Parametry mobilního připojení

```
-----  
Zabezpečení : enabled  
Limit mezipaměti vazeb : 32  
Funkce odpovídajícího uzlu (CN) : enabled
```

Tab. 3.: Seznam voleb mobility systému Windows XP

Volby	Význam
security	zabezpečení aktualizace vazeb (boolean)
bindingcachelimit	maximální počet položek v mezipaměti vazeb (číslo)
correspondentnode	podpora korespondenta (boolean)

3.3 Ověření implementace

Na základě konfigurace popsané v předchozích kapitolách jsem provedl praktické ověření mobilní implementace na obou platformách. Testování jsem rozdělil z hlediska funkce mobility na následující oblasti:

- domáci registrace
 - získání domácího agenta
 - návrat mobilního uzlu do domácí sítě
- optimalizace cesty
 - optimalizace směrování při přenosu dat mezi mobilním uzlem a korespondentem

Nejdříve přišla na řadu hlavní testovací platforma Linux, kde došlo k otestování obou zvolených oblastí.

3.3.1 Linux

Pro ověření funkcionality mobility a vizualizaci zkoumaných dějů do použitelné podoby jsem využil několika příkazů/nástrojů:

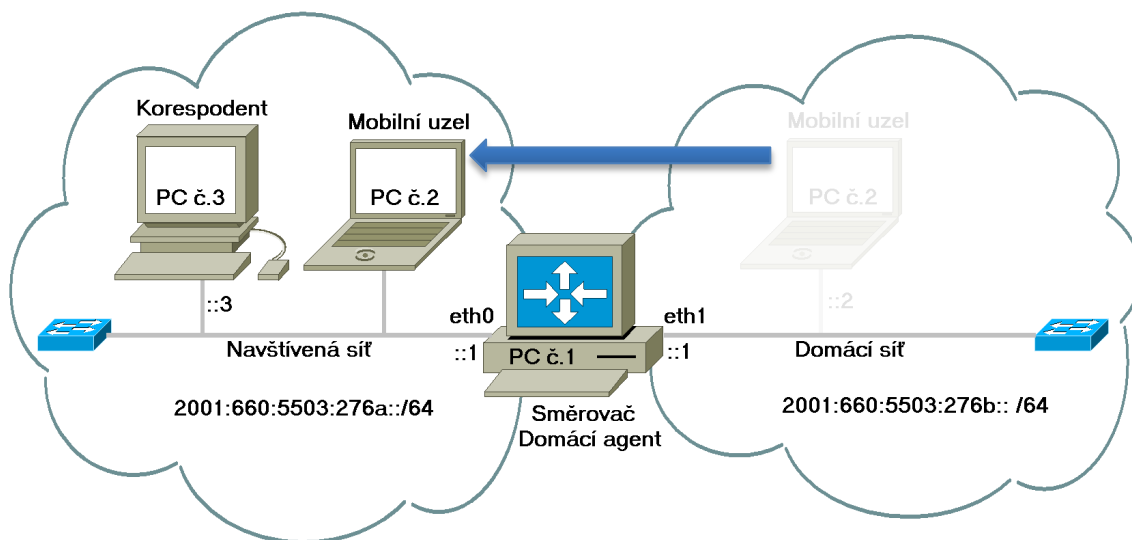
- *ifconfig* – zobrazí informace o síťových rozhraních na daném počítači
- *telnet localhost 7777* – zpřístupní další informace o aktuálním stavu mobility
 - *bc* – vypíše údaje z cache vazeb (domácí agent, korespondent)⁹
 - *bul* – vypíše seznam zaslaných aktualizací vazeb (mobilní uzel)⁹
- *route -A inet6* – zobrazí směrovací tabulku protokolu IPv6
- *ping6 [IPv6 adresa]* – odesílání požadavku na odezvu
- *aplikace Ethereal*¹⁰ – sledování a analýza síťové komunikace

⁹ Příkaz má význam pouze pro roli mobility uvedenou v závorce

¹⁰ Aplikace byla v systému Knoppix již nainstalována. Jde o předchůdce aplikace Wireshark

Domácí registrace – získání domácího agenta

Při tomto testu se mobilní uzel přesunul z výchozí ustálené pozice v domácí síti do navštívené sítě, což také ilustruje Obr. 3..



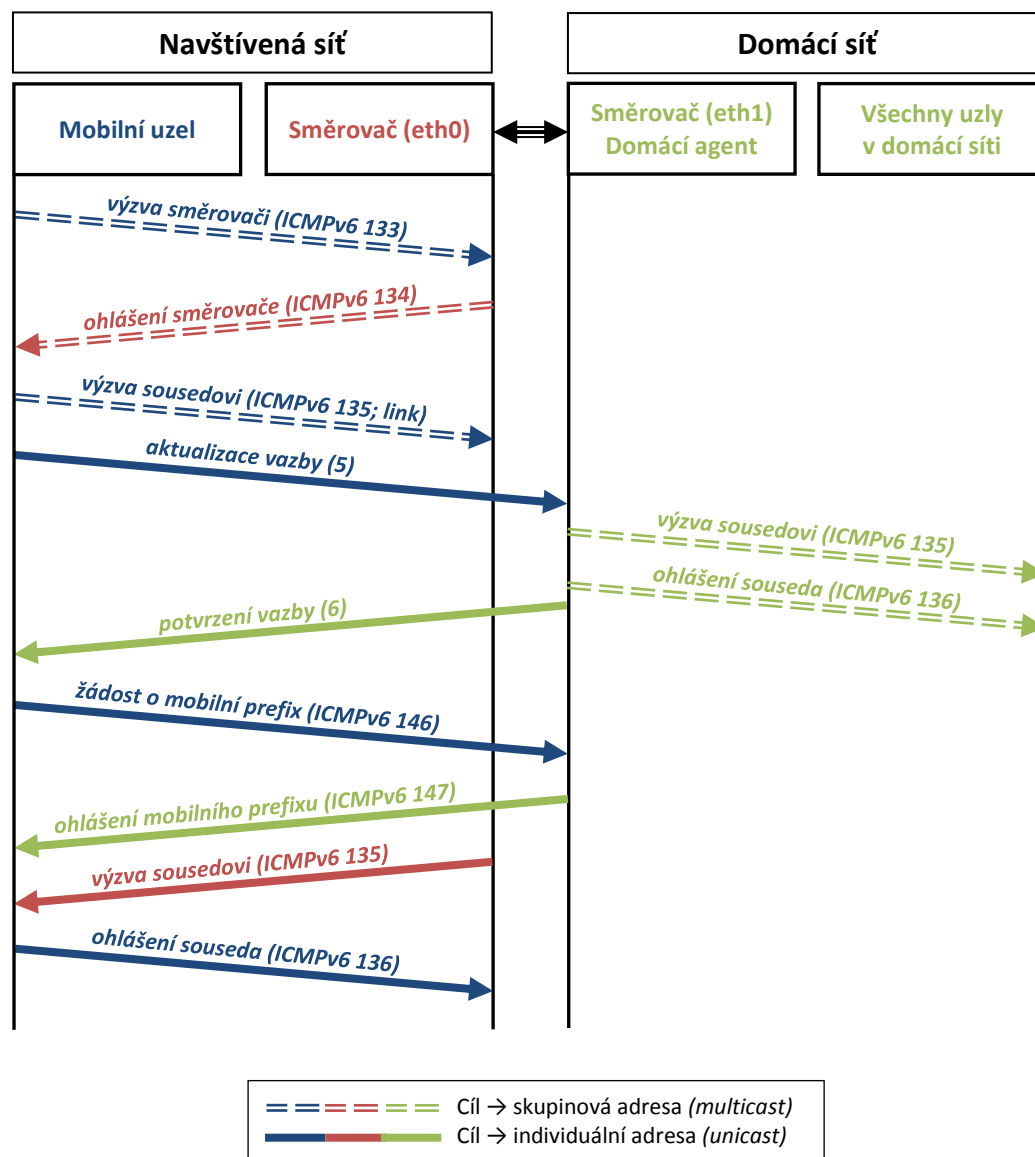
Obr. 3.: Výchozí stav v síti – test domácí registrace

Okamžik připojení mobilního uzlu do návštěvní sítě a následnou komunikaci mezi jednotlivými síťovými prvky jsem sledoval a analyzoval aplikací Ethereal.

Po přepojení do navštívené sítě si mobilní uzel nejdříve výzvou směrovači (*ICMPv6 133*) vyžádal informace o místních směrovačích, načež dostal odpověď od jediného směrovače v síti v podobě ohlášení směrovače (*ICMPv6 134*). Mobilní uzel si také vygeneroval svou lokální linkovou adresu a výzvou sousedovi (*ICMPv6 135*) kontroloval její případnou duplicitu. Na základě informací z ohlášení směrovače si už vytvořil i svou dočasnou adresu, a nic mu tedy nebránilo v zaslání aktualizace vazby (5) na adresu domácího agenta. Ten mu na oplátku odeslal potvrzení vazby (6), ale mezitím ještě zkontroloval výzvou sousedovi (*ICMPv6 135*) případnou duplicitu domácí adresy mobilního uzlu v domácí síti. Nakonec domácí agent odeslal na skupinovou adresu (ff02::1) nevyžádané ohlášení souseda (*ICMPv6 136*), čímž si zajistil přesměrování datagramů určených pro mobilní uzel na sebe a tím úspěšně zakončil proces domácí registrace.

Poté se ještě směrovač v navštívené síti dotázal výzvou sousedovi (ICMPv6 135) na linkovou adresu mobilního uzlu, který mu ji sdělil zprávou ohlášení souseda (ICMPv6 136).

Schématické zobrazení popsané výměny datagramů mezi jednotlivými síťovými prvky ukazuje Obr. 3.3. Původní podobu výstupu z aplikace Ethereal v textové podobě lze nalézt v příloze E.



Obr. 3.: Znázornění výměny datagramů – domácí registrace mobilního uzlu

Dále jsem zjišťoval, jakým způsobem se změnila konfigurace síťového rozhraní mobilního uzlu při jeho přesunu z domácí sítě do navštívené:

✓ stav rozhraní mobilního uzlu – domácí síť

```
eth0      Link encap:Ethernet  HWaddr 00:17:08:30:02:D4
          inet6 addr: 2001:660:5503:276b::2/64 Scope:Global
          inet6 addr: 2001:660:5503:276b:217:8ff:fe30:2d4/64
Scope:Global
          inet6 addr: fe80::217:8ff:fe30:2d4/64 Scope:Link

ip6tnl1   Link encap:UNSPEC  HWaddr 20-01-06-60-55-03-27-6B-00-
00-00-00-00-00-00-00-00
          inet6 addr: fe80::217:8ff:fe30:2d4/64 Scope:Link
```

✓ stav rozhraní mobilního uzlu – navštívená síť

```
eth0      Link encap:Ethernet  HWaddr 00:17:08:30:02:D4
          inet6 addr: 2001:660:5503:276a:217:8ff:fe30:2d4/64
Scope:Global
          inet6 addr: fe80::217:8ff:fe30:2d4/64 Scope:Link

ip6tnl1   Link encap:UNSPEC  HWaddr 20-01-06-60-55-03-27-6A-00-
00-00-00-00-00-00-00-00
          inet6 addr: 2001:660:5503:276b::2/128 Scope:Global
          inet6 addr: fe80::217:8ff:fe30:2d4/64 Scope:Link
```

Z výpisů je patrné, že dokud mobilní uzel sídlil v domácí síti, byl dostupný na své domácí adrese (*2001:660:5503:276b::2/64*). Po přepojení do navštívené sítě si pak vygeneroval novou dočasnou adresu (*2001:660:5503:276a:217:8ff:fe30:2d4/64*). V rámci tunelového spojení s domácím agentem (rozhraní *ip6tnl1*) si mobilní uzel stále udržoval svou původní domácí adresu. Tunelové rozhraní *ip6tnl1* se po úspěšné domácí registraci automaticky vytvořilo i na straně domácího agenta:

✓ stav rozhraní *ip6tnl1* směrovače / domácího agenta (mobilní uzel v navštívené síti)

```
ip6tnl1   Link encap:UNSPEC  HWaddr 20-01-06-60-55-03-27-6B-00-
00-00-00-00-00-00-00-00
          inet6 addr: fe80::20e:2eff:fe6c:c0b5/64 Scope:Link
```

Rovněž jsem sledoval změny z hlediska mobility ve směrovacích tabulkách (kompletní výpis záznamů je možné nalézt v příloze D):

✓ směrovací tabulka mobilního uzlu – navštívená síť

```
Kernel IPv6 routing table
Destination                                Next Hop                                Iface
::/0                                       ::                                     ip6tnl1
2001:660:5503:276b::1/128              2001:660:5503:276b::1              ip6tnl1
2001:660:5503:276a::/64                  ::                                    eth0
....
```

Po přepojení mobilního uzlu do navštívené sítě se v jeho směrovací tabulce vytvořila nová implicitní cesta (::/0) směřující do tunelového rozhraní *ip6tnl1*. Druhý záznam pak přímo specifikuje cestu k domácímu agentovi (2001:660:5503:276b::1/128) opět přes tunelové rozhraní *ip6tnl1*.

Pro doručování datagramů tunelem v druhém směru, od domácího agenta k mobilnímu uzlu, vznikl také ve směrovací tabulce domácího agenta odpovídající záznam:

✓ směrovací tabulka směrovače / domácí agenta (mobilní uzel v navštívené síti)

```
Kernel IPv6 routing table
Destination                Next Hop                    Iface
2001:660:5503:276b::2/128  ::                          ip6tnl1
2001:660:5503:276a::/64    ::                          eth0
2001:660:5503:276b::/64    ::                          eth1
....
```

Po úspěšné domácí registraci by měl mít domácí agent vytvořený záznam s údaji o mobilním uzlu v datové struktuře – cache vazeb. Mobilní uzel by si zase měl vést datovou strukturu – seznam aktualizací vazeb s údaji o uzlech, kterým poslal aktualizaci vazby. Tento předpoklad se během testu potvrdil. Níže uvádím výpis obou struktur:

✓ výpis cache vazeb – domácí agent

```
mip6d> bc
  hoa 2001:660:5503:276b:0:0:0:2 status registered
  coa 2001:660:5503:276a:217:8ff:fe30:2d4 flags AH--
  local 2001:660:5503:276b:0:0:0:1
  lifetime 11541 / 11988 seq 22451 unreach 0 mpa 360 / 805
```

Ze záznamu lze vyčíst informace o domácí adrese (*hoa*) a dočasné adrese (*coa*) mobilního uzlu. Příznak AH (*flags AH*) prozrazuje, že se jedná domácí registraci. Záznam dále obsahuje lokální adresu domácího uzlu (*local*), životnost (*lifetime*) a pořadové číslo (*seq*).

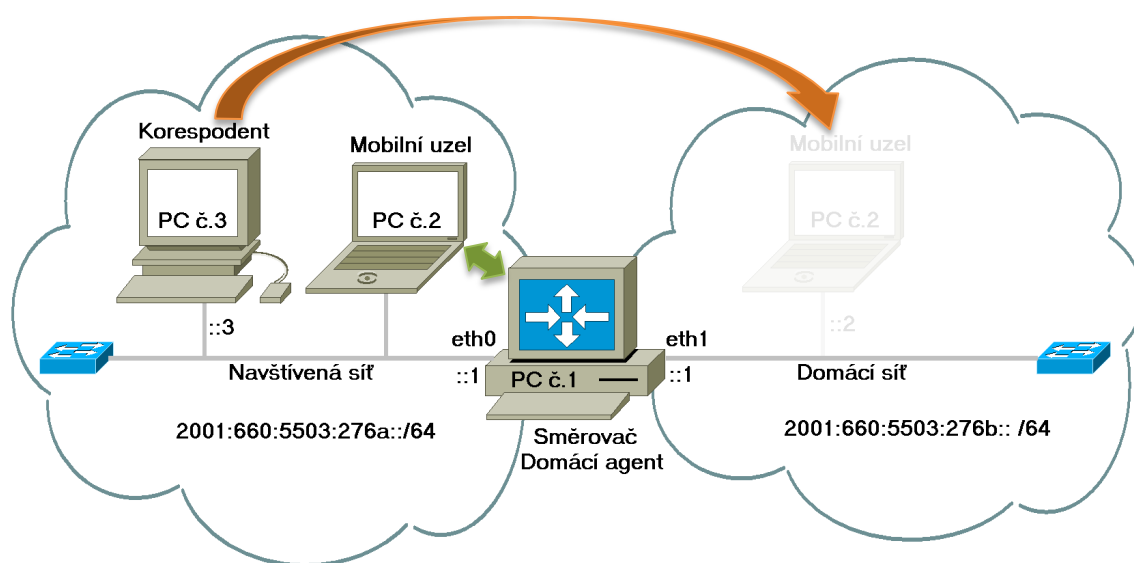
✓ výpis seznamu aktualizací vazeb – mobilní uzel

```
mip6d> bul
  hoa 2001:660:5503:276b:0:0:0:2 cn 2001:660:5503:276b:0:0:0:1
  coa 2001:660:5503:276a:217:8ff:fe30:2d4 flags AH-- type bul ack
ready
  lifetime 11809 / 11984 seq 53357 resend 0 delay 11384(after
11209s)
  mps 10626 / 10798
```

Kromě údajů shodných s předchozím záznamem obsahuje záznam ze seznamu aktualizací navíc adresu korespondenta (*cn*), v tomto případě domácího agenta, kterému mobilní uzel zaslal aktualizaci vazby.

Optimalizace cesty

V rámci testu optimalizace cesty se mobilní uzel již od počátku nacházel v navštívené síti a měl ustanovenou vazbu s domácím agentem. Korespondent se také nacházel v navštívené síti a zahájil komunikaci s mobilním uzlem přes jeho domácí adresu, kterou měl jako jedinou k dispozici. Tento stav v síti ukazuje Obr. 3..



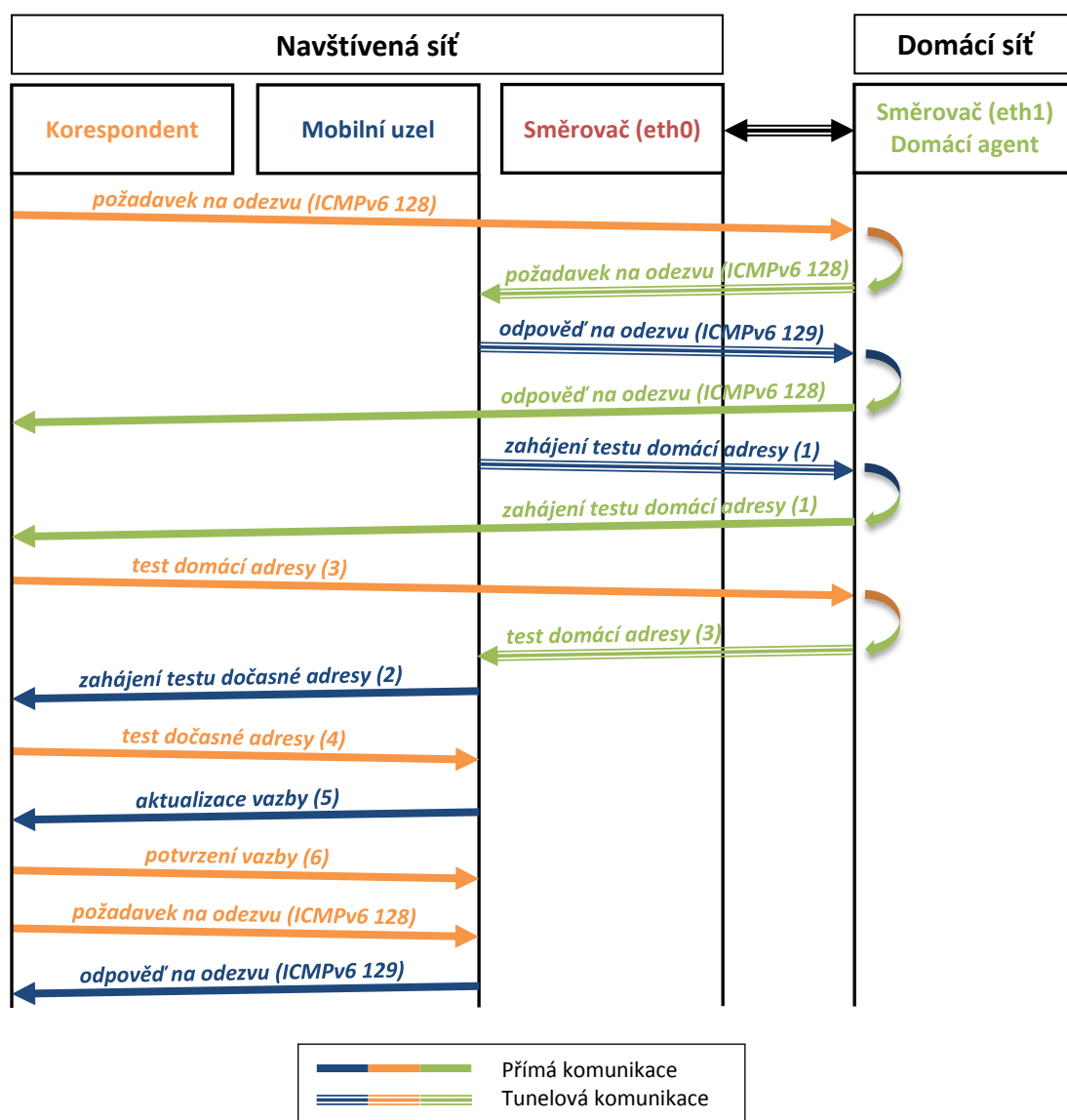
Obr. 3.: Výchozí stav v síti – test optimalizace cesty

Celou síťovou komunikaci jsem opět sledoval pomocí aplikace Ethereal. Pro simulaci komunikace jsem využil příkaz *ping6* generující požadavek na odezvu.

Korespondent zahájil komunikaci zasláním požadavku na odezvu (*ICMPv6 128*) na domácí adresu mobilního uzlu. Zprávu přijal domácí agent zastupující mobilní uzel na jeho domácí adrese a přeposlal ji tunelem mobilnímu uzlu na jeho dočasnou adresu. Ten na přeposlanou zprávu reagoval odpovědí na odezvu (*ICMPv6 129*), kterou odeslal tunelem domácímu agentovi, jenž ji následně přeposlal korespondentovi. Zároveň s touto odpovědí poslal mobilní uzel stejnou cestou (přes domácího agenta) korespondentovi zahájení testu domácí adresy (1). Korespondent zprávu přijal a na oplátku odeslal mobilnímu uzlu test domácí adresy (3) opět cestou přes domácího agenta. Následně mobilní uzel odeslal zahájení testu dočasné adresy (2), tentokrát už přímo na adresu kore-

spondenta, kterou se dozvěděl hned z první zprávy přeposlané od domácího agenta. Ko-respondent se dozvěděl dočasnou adresu mobilního uzlu teprve ze zprávy zahájení testu dočasné adresy, na kterou reagoval odesláním testu dočasné adresy (4). Dále již mobilní uzel na základě zpráv (3;4) odeslal korespondentovi aktualizaci vazby (5). Ten odpověděl potvrzením vazby (6) a zakončil tak úspěšně celý proces optimalizace cesty.

Přehledné grafické ztvárnění výše popsaného procesu je možné vidět na Obr. 3.5. Původní textová verze výstupu z aplikace Ethereal se nachází v příloze E.



Obr. 3.: Znázornění výměny datagramů – optimalizace cesty

Konfigurace síťových rozhraní uzlů se při tomto testu nijak nezměnila. Z hlediska mobility zaujme směrovací tabulka mobilního uzlu (kompletní tabulky všech uzlů jsou umístěny v příloze D):

✓ směrovací tabulka mobilního uzlu – komunikace s korespondentem

```
Kernel IPv6 routing table
Destination                                Iface
Next Hop
2001:660:5503:276a:2e0:4cff:fedb:1a16/128  ip6tnl1
2001:660:5503:276a:2e0:4cff:fedb:1a16
.....
2001:660:5503:276a:2e0:4cff:fedb:1a16/128  eth0
2001:660:5503:276a:2e0:4cff:fedb:1a16
.....
```

Oba záznamy určují cestu ke korespondentovi. První vznikl ještě před provedením optimalizace cesty, kdy se všechny datagramy ke korespondentovi musely posílat tunelem domácímu agentovi. Druhý záznam již značí přímou cestu ke korespondentovi přes rozhraní *eth0*.

Nakonec jsem obdobně jako v minulém testu ověřoval záznamy v cache vazeb, tentokrát však na straně korespondenta, a v seznamu aktualizací vazeb mobilního uzlu:

✓ výpis cache vazeb – korespondent

```
mip6d> bc
  hoa 2001:660:5503:276b:0:0:0:2 status cached
  coa 2001:660:5503:276a:217:8ff:fe30:2d4 flags A---
  local 2001:660:5503:276a:2e0:4cff:fedb:1a16
  lifetime 95 / 420 seq 48194 unreach 0
```

Záznam obsahuje všechny potřebné údaje o mobilním uzlu. Oproti záznamu z cache vazeb domácího agenta pochopitelně chybí příznak H, protože se nejedná o domácí registraci.

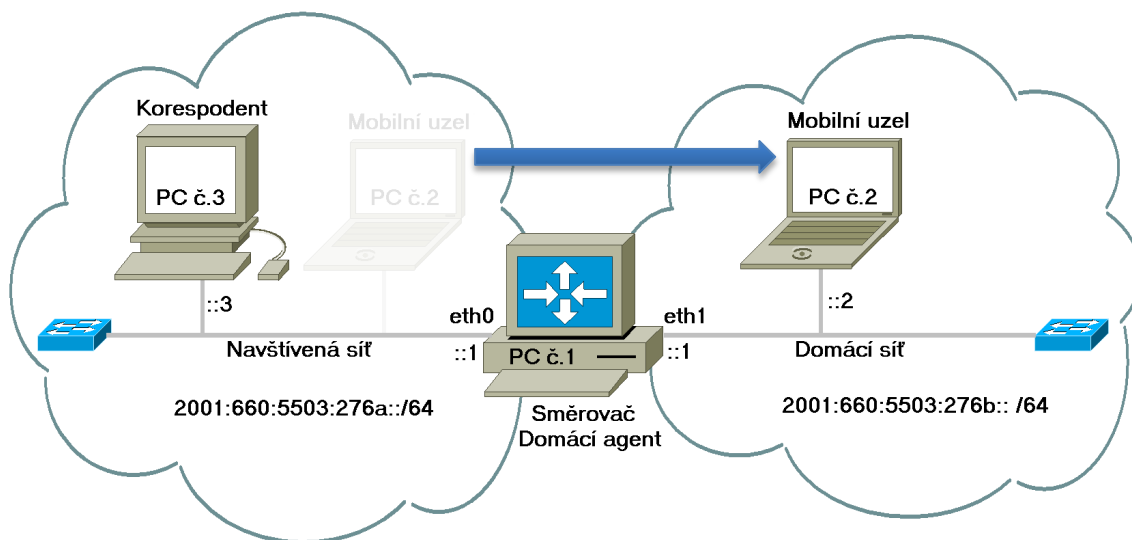
✓ výpis seznamu aktualizací vazeb – mobilní uzel

```
mip6d> bul
  hoa 2001:660:5503:276b:0:0:0:2 cn 2001:660:5503:276b:0:0:0:1
  coa 2001:660:5503:276a:217:8ff:fe30:2d4 flags AH-- type bul
ack ready
  lifetime 11799 / 11988 seq 62405 resend 0 delay 11388(after
11200s)
  mps 10612 / 10798
  hoa 2001:660:5503:276b:0:0:0:2 cn
2001:660:5503:276a:2e0:4cff:fedb:1a16
  coa 2001:660:5503:276a:217:8ff:fe30:2d4 flags A--- type bul
ack ready RR state ready
  lifetime 234 / 420 seq 48194 resend 0 delay 420(after 235s)
```

Seznam aktualizací vazeb mobilního uzlu se dle očekávání rozšířil o nový záznam (zvýrazněn tučným písmem) nesoucí informace o korespondentovi.

Návrat mobilního uzlu domů

Závěrečný test na platformě Linux spočíval v návratu mobilního uzlu zpět do své domácí sítě z výchozí ustálené pozice v navštívené síti. Tento stav znázorňuje Obr. 3..

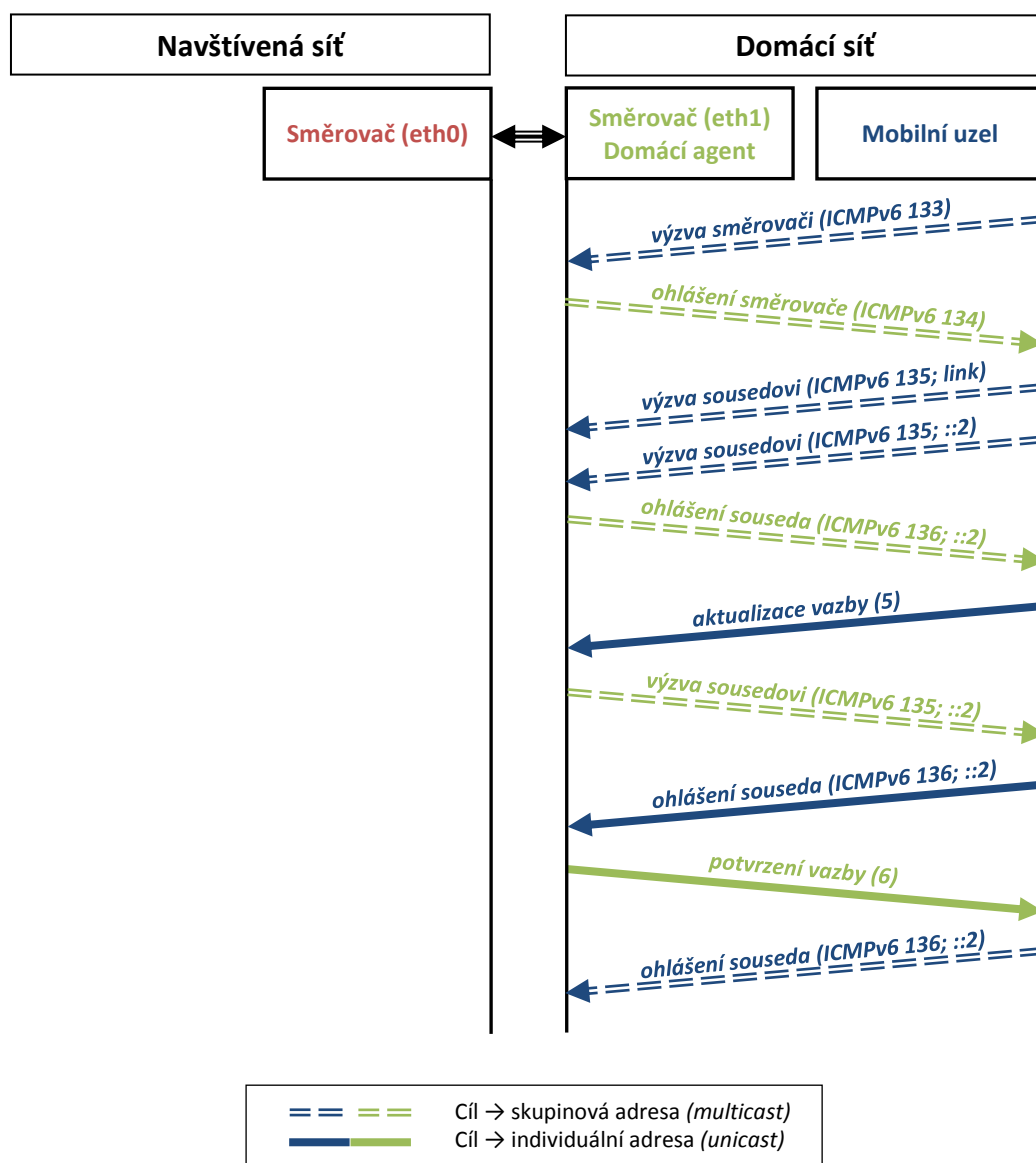


Obr. 3.: Výchozí stav v síti – návrat mobilního uzlu domů

Po přepojení do domácí sítě si mobilní uzel standardně výzvou směrovači (*ICMPv6 133*) vyžádal informace o místních směrovačích. Na zprávu zareagoval „domácí“ směrovač ohlášením směrovače (*ICMPv6 134*). Z tohoto ohlášení již mobilní uzel poznal, že je v domácí síti, a začal zjišťovat výzvou sousedovi (*ICMPv6 135*), jak to vypadá s jeho lokální linkovou adresou a domácí adresou v síti. Na domácí adrese mobilního uzlu stále sídlil zastupující domácí agent, který tedy zareagoval ohlášením souseda (*ICMPv6 136*). Dále již mobilní uzel zaslal aktualizaci vazby (5) s nulovou životností a příznakem „H“, čímž dal domácímu agentovi najevo, že se vrátil domů a požaduje zrušení domácí registrace. Na další výzvu sousedovi (*ICMPv6 135*) od domácího agenta reagoval mobilní uzel ohlášením souseda (*ICMPv6 136*), kde uvádí svou domácí a linkovou adresu a příznakem „O“, čímž de facto domácího agenta zbavil funkce svého zástupce. Potvrzením vazby (6) s nulovou životností domácí agent informoval, že vymazal cache vazeb a zrušil tak domácí registraci. Konečně nevyžádaným

ohlášením souseda (*ICMPv6 136*) zaslaným na skupinovou adresu sdělil mobilní uzel i všem ostatním uzlům v síti, že přebírá zpět svůj síťový provoz.

Celý výše popsany proces graficky znázorňuje Obr. 3.7. Původní výpis z aplikace Ethereal je k dispozici v příloze E.



Obr. 3.: Znázornění výměny datagramů – návrat mobilního uzlu domů

Mobilní uzel získal zpět svou domácí adresu a tunelové spojení s domácím agentem se po zrušení domácí registrace ukončilo:

✓ stav rozhraní mobilního uzlu – domácí síť

```
eth0      Link encap:Ethernet  HWaddr 00:17:08:30:02:D4
          inet6 addr: 2001:660:5503:276b::2/64 Scope:Global
          inet6 addr: 2001:660:5503:276b:217:8ff:fe30:2d4/64
Scope:Global
          inet6 addr: fe80::217:8ff:fe30:2d4/64 Scope:Link

ip6tnl1   Link encap:UNSPEC  HWaddr 20-01-06-60-55-03-27-6B-00-
00-00-00-00-00-00-00-00
          inet6 addr: fe80::217:8ff:fe30:2d4/64 Scope:Link
```

S ukončením tunelového spojení se zrušilo i související směrování. Také došlo k vymazání cache vazeb u domácího agenta a seznamu aktualizací vazeb u mobilního uzlu:

✓ výpis cache vazeb – domácí agent

```
mip6d> bc
....
```

✓ výpis seznamu aktualizací vazeb – mobilní uzel

```
mip6d> bul
....
```

3.3.2 MS Windows

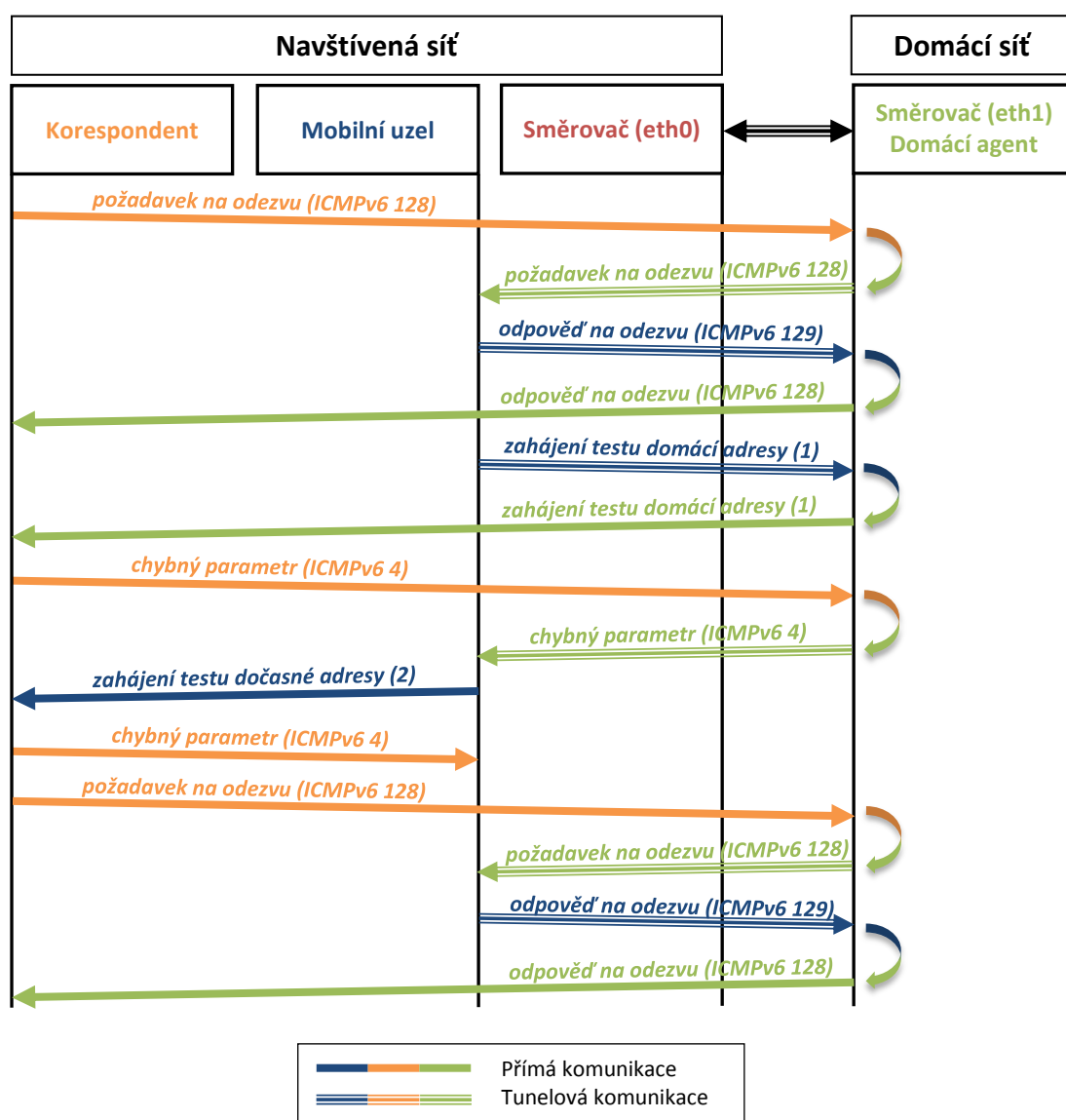
Na platformě MS Windows bylo možné vyzkoušet jen druhou oblast testování, která souvisí s rolí korespondenta a procesem optimalizace cesty. Na počítači č.3 byl spuštěn systém Windows XP a nastaven pro roli korespondenta. Konfigurace ostatních počítačů zůstala od předešlých testů beze změny.

Pro zachycení datagramů mezi počítači jsem opět použil aplikaci Ethereal a pro simulaci síťové komunikace příkaz *ping6* generující požadavek na odezvu. Na počátku testu se již mobilní uzel nacházel v navštívené síti a měl ustanovenou vazbu s domácím agentem. Korespondent se taktéž nacházel v navštívené síti a zahájil komunikaci s mobilním uzlem s použitím jeho domácí adresy. Výchozí stav v síti znázorňuje Obr. 3..

Začátek síťové komunikace probíhal stejným způsobem jako v testu optimalizace cesty na platformě Linux. Korespondent odeslal požadavek na odezvu (*ICMPv6 128*) na domácí adresu mobilního uzlu. Zastupující domácí agent požadavek přeposlal tune-

lem mobilnímu uzlu na jeho dočasnou adresu. Mobilní uzel odeslal tunelem odpověď na odezvu (*ICMPv6 129*) domácímu agentovi, který ji následně přeposlal korespondentovi. Zároveň mobilní uzel odeslal tunelem i zahájení testu domácí adresy (1). Korespondent však tuto zprávu nerozpoznal a namísto odeslání testu domácí adresy (3) poslal zprávu – chybný parametr (*ICMPv6 4*). Obdobně reagoval i na zahájení testu dočasné adresy (2). Z tohoto důvodu nemohl mobilní uzel poslat aktualizaci vazby (5) a optimalizace cesty se tak vůbec neuskutečnila. Další síťová komunikace musela probíhat stejně jako zpočátku, tedy cestou přes domácího agenta.

Popsanou výměnu datagramů je možné vidět na Obr. 3.8. Původní textový výstup z aplikace Ethereal lze nalézt v příloze E.



Obr. 3.: Znázornění výměny datagramů – optimalizace cesty (MS Windows)

4 Závěr

Výzkum konaný v rámci této bakalářské práce potvrdil, že mobilní IPv6 lze využít k vytvoření stabilního spojení s potencionálně se přemísťujícím zařízením. V této době však není k dispozici mnoho programů pro podporu této funkcionality.

Nativní podpora mobility se v současných systémech prakticky nevyskytuje. Novější verze systému MS Windows ji zcela ignorují. Starší systém Windows XP, který jsem zařadil do testů, sice jistou podporu mobility sliboval, ale reálná funkčnost se bohužel nepotvrdila.

Distribuce Linuxu taktéž nativní podporu mobility neobsahují. Vzniklo sice několik projektů, které se implementací mobility do systému zabývaly, nicméně většina z nich již není příliš aktuální. Jako nejpraktičtější varianta pro testování se jevila upravená verze live distribuce Knoppix s implementovanou podporou mobility vytvořenou pod záštitou Evropského ústavu pro telekomunikační normy.

Domnívám se, že za současnou neuspokojivou situaci ohledně podpory mobility vězí prozatím malá rozšířenost protokolu IPv6, a věřím, že s jeho postupným rozšiřováním se bude zlepšovat i podpora mobility v moderních operačních systémech.

Výsledky testování mobilní implementace na platformě Linux se shodovaly s principy popsány v teoretickém úvodu. S trochou nadsázky lze říct, že mě během testování na této platformě nic nečekaného nepřekvapilo. Díky aplikaci Ethereal jsem mohl v reálném čase sledovat výměnu datagramů mezi uzly, například při domácí registraci, a analyzovat hlavičky jednotlivých zpráv. Výpisy získané z této aplikace pak sloužily jako podklad pro tvorbu grafického ztvárnění sledovaného procesu.

Testování role korespondenta, potažmo procesu optimalizace cesty na platformě MS Windows, se nesetkalo s pozitivním výsledkem. Systém Windows XP nedokázal zpracovat zprávy s hlavičkou mobility, a celá komunikace tak musela být stále neefektivně směrována přes domácího agenta. Na druhou stranu jsem díky tomu získal nové výsledky i k této variantě komunikace.

Za velkou výhodu mobility považuji kromě zajištění vlastní dostupnosti „ať jsem kdekoli“, také v zprostředkování nepřerušované komunikace. Během testů jsem několikrát přepojoval mobilní uzel z jedné sítě do druhé za současně probíhající komunikace s korespondentem a ani v jednom případě nedošlo k totálnímu výpadku spojení

či ztrátě datagramů. Během přepojování se logicky komunikace zastaví, ale po zapojení mobilního uzlu do nové sítě a odeslání aktualizace vazby se komunikace ihned obnoví.

Celkově vzato považuji mobilní implementaci v protokolu IPv6 za velmi povedenou a užitečnou funkcionalitu a nedivil bych se, kdyby se v budoucnu stala běžnou součástí „online“ života.

Seznam použité literatury

- [1] **Kállay, Fedor a Peniak, Peter.** *Počítačové sítě a jejich aplikace LAN/MAN/WAN*. Druhé, aktualizované vydání. Praha : Grada Publishing, a.s., 2003. str. 356. ISBN 80-247-0545-1.
- [2] **Li, Qing, Jinmei, Tatuya a Shima, Keiichi.** *Mobile IPv6: Protocols and Implementation*. Burlington, USA : Morgan Kaufmann Publishers, 2009. str. 467. ISBN 978-0-12-375075-4.
- [3] **Satrapa, Pavel.** *Internetový protokol verze 6*. Třetí, aktualizované a doplněné vydání. Praha : CZ.NIC, z.s.p.o., 2011. str. 407. ISBN 978-80-904248-4-5.
- [4] **kolektiv autorů.** *An IPv6 Deployment Guide*. 6NET. [Online] 2005. <http://www.6net.org/book/deployment-guide.pdf>.
- [5] **kolektiv autorů.** *Mobile IPv6 Mini Howto*. TC MTS-IPT: IPv6 Testing an eEurope Project. [Online] http://www.ipt.etsi.org/mini_howto.htm.
- [6] **kolektiv autorů.** *Linux Mobile IPv6 HOWTO*. [Online] 2004. <http://tldp.org/HOWTO/Mobile-IPv6-HOWTO/>.
- [7] **kolektiv autorů.** *Enabling efficient and operational mobility in large heterogeneous*. [Online] 2008. <http://www.ipv6tf.org/pdf/enablebook.pdf>. ISBN 978-84-691-0647-1.
- [8] **kolektiv autorů.** *IPv6*. [Online] <http://www.ipv6.cz>.
- [9] **Internet Engineering Task Force (IETF).** *Mobility Support in IPv6*. RFC 6275. [Online] 2011. <http://tools.ietf.org/html/rfc6275>. ISSN 2070-1721.
- [10] **Jirovský, Václav.** *Vademecum správce sítě*. Praha : Grada Publishing, spol. s.r.o., 2001. str. 428. ISBN 80-7169-745-1.
- [11] **Pužmanová, Rita a Šmrha, Pavel.** *Propojování sítí s TCP/IP*. České Budějovice : nakladatelství KOPP, 1999. str. 203. ISBN 80-7232-080-7.

Přílohy

A. Výpis programu radvdump

```
# radvd configuration generated by radvdump 0.9.1
# based on Router Advertisement from fe80::20e:2eff:fe6c:c0b5
# received by interface eth0
#

interface eth0
{
    AdvSendAdvert on;
    # Note: {Min,Max}RtrAdvInterval cannot be obtained with
radvdump
    AdvManagedFlag off;
    AdvOtherConfigFlag off;
    AdvReachableTime 0;
    AdvRetransTimer 0;
    AdvCurHopLimit 64;
    AdvHomeAgentFlag off;
    AdvDefaultPreference medium;
    AdvSourceLLAddress on;
    AdvIntervalOpt on;

    prefix 2001:660:5503:276a::/64
    {
        AdvValidLifetime 2592000;
        AdvPreferredLifetime 604800;
        AdvOnLink on;
        AdvAutonomous on;
        AdvRouterAddr on;
    }; # End of prefix definition
}; # End of interface definition
#
# radvd configuration generated by radvdump 0.9.1
# based on Router Advertisement from fe80::2e0:18ff:fe43:687
# received by interface eth1
#
```

```

interface eth1
{
    AdvSendAdvert on;
    # Note: {Min,Max}RtrAdvInterval cannot be obtained with
radvdump
    AdvManagedFlag off;
    AdvOtherConfigFlag off;
    AdvReachableTime 0;
    AdvRetransTimer 0;
    AdvCurHopLimit 64;
    AdvHomeAgentFlag on;
    AdvDefaultPreference medium;
    AdvSourceLLAddress on;
    AdvIntervalOpt on;
    AdvHomeAgentInfo on;
    AdvMobRtrSupportFlag off;
    HomeAgentPreference 20;
    HomeAgentLifetime 10000;

    prefix 2001:660:5503:276b::/64
    {
        AdvValidLifetime 12000;
        AdvPreferredLifetime 10000;
        AdvOnLink on;
        AdvAutonomous on;
        AdvRouterAddr on;
    }; # End of prefix definition
}; # End of interface definition

```

B. Výpis konfigurace síťových rozhraní

Počítač č.1 (směrovač / domácí agent)

```
root@0[~]# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0E:2E:6C:C0:B5
          inet6 addr: 2001:660:5503:276a::1/64 Scope:Global
          inet6 addr: fe80::20e:2eff:fe6c:c0b5/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:167 errors:0 dropped:0 overruns:0 frame:0
          TX packets:30773 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:15070 (14.7 KiB)  TX bytes:3629922 (3.4 MiB)
          Interrupt:11 Base address:0x8000

eth1      Link encap:Ethernet  HWaddr 00:E0:18:43:06:87
          inet6 addr: 2001:660:5503:276b::1/64 Scope:Global
          inet6 addr: fe80::2e0:18ff:fe43:687/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:260 errors:0 dropped:0 overruns:0 frame:0
          TX packets:31322 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:22828 (22.2 KiB)  TX bytes:3938344 (3.7 MiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:860 errors:0 dropped:0 overruns:0 frame:0
          TX packets:860 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:46783 (45.6 KiB)  TX bytes:46783 (45.6 KiB)
```

Počítač č.2 (mobilní uzel)

```
root@0[~]# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:17:08:30:02:D4
          inet6 addr: 2001:660:5503:276b::2/64 Scope:Global
          inet6 addr: 2001:660:5503:276b:217:8ff:fe30:2d4/64
Scope:Global
          inet6 addr: fe80::217:8ff:fe30:2d4/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:31896 errors:0 dropped:0 overruns:0 frame:0
          TX packets:577 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:3959160 (3.7 MiB)  TX bytes:51778 (50.5 KiB)
          Interrupt:16
```



```

ip6tnl1  Link encap:UNSPEC  HWaddr 20-01-06-60-55-03-27-6B-00-
00-00-00-00-00-00-00
        inet6 addr: fe80::217:8ff:fe30:2d4/64 Scope:Link
        UP POINTOPOINT RUNNING NOARP  MTU:1460  Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0
        RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)

lo        Link encap:Local Loopback
        inet addr:127.0.0.1  Mask:255.0.0.0
        inet6 addr: ::1/128 Scope:Host
        UP LOOPBACK RUNNING  MTU:16436  Metric:1
        RX packets:388 errors:0 dropped:0 overruns:0 frame:0
        TX packets:388 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0
        RX bytes:21062 (20.5 KiB)  TX bytes:21062 (20.5 KiB)

```

Počítač č.3 (korespondent)

```

root@0[~]# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:E0:4C:DB:1A:16
        inet6 addr: 2001:660:5503:276a::3/64 Scope:Global
        inet6 addr: 2001:660:5503:276a:2e0:4cff:fedb:1a16/64
Scope:Global
        inet6 addr: fe80::2e0:4cff:fedb:1a16/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:486 errors:0 dropped:0 overruns:0 frame:0
        TX packets:130 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:56908 (55.5 KiB)  TX bytes:13996 (13.6 KiB)
        Interrupt:16 Base address:0x2c00

lo        Link encap:Local Loopback
        inet addr:127.0.0.1  Mask:255.0.0.0
        inet6 addr: ::1/128 Scope:Host
        UP LOOPBACK RUNNING  MTU:16436  Metric:1
        RX packets:63 errors:0 dropped:0 overruns:0 frame:0
        TX packets:63 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0
        RX bytes:4476 (4.3 KiB)  TX bytes:4476 (4.3 KiB)

```

C. Konfigurace mobilní implementace – varianta IPsec

Počítač č.1 (domácí agent)

- *konfigurační soubor /usr/local/etc/mip6d.conf*

```
# mip6d.conf : A basic Home Agent

NodeConfig HA;
DebugLevel 10;
DoRouteOptimizationCN enabled;

Interface "eth1";
UseMnHaIPsec enabled;

IPsecPolicySet {
    HomeAgentAddress 2001:660:5503:276b::1;
    HomeAddress 2001:660:5503:276b::2/64;

    IPsecPolicy HomeRegBinding UseESP;
    IPsecPolicy TunnelMh UseESP;
}
```

- *konfigurační soubor /usr/local/etc/sa.conf*

```
# sa.conf

# MN = 2001:660:5503:276b::2
# HA = 2001:660:5503:276b::1

# #-----
# # IPsec MN -> HA Transport mode (BU)
# #-----
add      2001:660:5503:276b::2
         2001:660:5503:276b::1
         esp 0001
         -m transport
         -E null
         -A null ;

#-----
# IPsec HA -> MN Transport mode (BA)
#-----
add      2001:660:5503:276b::1
         2001:660:5503:276b::2
         esp 0002
         -m transport
         -E null
         -A null ;
```

```

#-----
# IPsec MN -> HA Tunnel mode (HoTI)
#-----
add      2001:660:5503:276b::2
         2001:660:5503:276b::1
         esp 0016
         -m tunnel
         -E null
         -A null ;

#-----
# IPsec HA -> MN Tunnel mode (HoT)
#-----
add      2001:660:5503:276b::1
         2001:660:5503:276b::2
         esp 0017
         -m tunnel
         -E null
         -A null ;

```

Počítač č.2 (mobilní uzel)

- *konfigurační soubor /usr/local/etc/mip6d.conf*

```

# mip6d.conf : A basic Mobile Node

NodeConfig MN;
DebugLevel 10;
DoRouteOptimizationCN enabled;

Interface "eth0";
UseMnHaIPsec enabled;

DoRouteOptimizationMN enabled;
UseCnBuAck enabled;

MnHomeLink "eth0" {
    HomeAgentAddress 2001:660:5503:276b::1;
    HomeAddress 2001:660:5503:276b::2/64;
}

IPsecPolicySet {
    HomeAgentAddress 2001:660:5503:276b::1;
    HomeAddress 2001:660:5503:276b::2/64;

    IPsecPolicy HomeRegBinding UseESP;
    IPsecPolicy TunnelMh UseESP;
}

```

- *konfigurační soubor /usr/local/etc/sa.conf*

```
# sa.conf

# MN = 2001:660:5503:276b::2
# HA = 2001:660:5503:276b::1

# #-----
# # IPsec MN -> HA Transport mode (BU)
# #-----
add      2001:660:5503:276b::2
         2001:660:5503:276b::1
         esp 0001
         -m transport
         -E null
         -A null ;

#-----
# IPsec HA -> MN Transport mode (BA)
#-----
add      2001:660:5503:276b::1
         2001:660:5503:276b::2
         esp 0002
         -m transport
         -E null
         -A null ;

#-----
# IPsec MN -> HA Tunnel mode (HoTI)
#-----
add      2001:660:5503:276b::2
         2001:660:5503:276b::1
         esp 0016
         -m tunnel
         -E null
         -A null ;

#-----
# IPsec HA -> MN Tunnel mode (HoT)
#-----
add      2001:660:5503:276b::1
         2001:660:5503:276b::2
         esp 0017
         -m tunnel
         -E null
         -A null ;
```

D. Výpisy směrovacích tabulek

Směrovač / Domácí agent

▪ mobilní uzel zapojen v domácí síti

Kernel IPv6 routing table		Next Hop	Flags	Metric	Ref	Use	Iface
Destination							
2001:660:5503:276a:217:8ff:fe30:2d4/128		2001:660:5503:276a:217:8ff:fe30:2d4	UC	0	70	1	eth0
2001:660:5503:276a::/64		::	U	256	1	0	eth0
2001:660:5503:276b:2/128		2001:660:5503:276b::2	UC	0	1	1	eth1
2001:660:5503:276b::/64		::	U	256	0	0	eth1
fe80::/64		::	U	256	0	0	eth0
fe80::/64		::	U	256	0	0	eth1
ff02::1/128		ff02::1	UC	0	753	1	eth0
ff02::1/128		ff02::1	UC	0	751	0	eth1
ff00::/8		::	U	256	0	0	eth0
ff00::/8		::	U	256	0	0	eth1
::1/128		::	U	0	0	2	lo
2001:660:5503:276a::/128		::	U	0	0	2	lo
2001:660:5503:276a::1/128		::	U	0	0	2	lo
2001:660:5503:276b::/128		::	U	0	0	2	lo
2001:660:5503:276b::1/128		::	U	0	103	2	lo
2001:660:5503:276b:fdff:ffff:ffff:fffe/128		::	U	0	0	2	lo
fe80::/128		::	U	0	0	2	lo
fe80::/128		::	U	0	0	2	lo
fe80::20e:2eff:fe6c:c0b5/128		::	U	0	33	2	lo
fe80::2e0:18ff:fe43:687/128		::	U	0	16	2	lo

▪ mobilní uzel zapojen v navštívené síti

Kernel IPv6 routing table		Next Hop	Flags	Metric	Ref	Use	Iface
Destination							
2001:660:5503:276b:2/128		2001:660:5503:276b::2	UC	0	1	1	eth1
2001:660:5503:276b:2/128		::	U	128	0	0	eth1
2001:660:5503:276b:2/128		::	U	192	0	0	ip6tnl1
2001:660:5503:276a:217:8ff:fe30:2d4/128		2001:660:5503:276a:217:8ff:fe30:2d4	UC	0	73	2	eth0
2001:660:5503:276a::/64		::	U	256	1	0	eth0
2001:660:5503:276b::/64		::	U	256	0	0	eth1
fe80::/64		::	U	256	0	0	eth0
fe80::/64		::	U	256	0	0	eth1
fe80::/64		::	U	256	0	0	ip6tnl1
ff02::1/128		ff02::1	UC	0	162	0	eth0
ff02::1/128		ff02::1	UC	0	162	1	eth1
ff00::/8		::	U	256	0	0	eth0
ff00::/8		::	U	256	0	0	eth1
ff00::/8		::	U	256	0	0	ip6tnl1
::1/128		::	U	0	0	2	lo
2001:660:5503:276a::/128		::	U	0	0	2	lo
2001:660:5503:276a::1/128		::	U	0	0	2	lo
2001:660:5503:276b::/128		::	U	0	0	2	lo
2001:660:5503:276b::1/128		::	U	0	105	2	lo
2001:660:5503:276b:fdff:ffff:ffff:fffe/128		::	U	0	0	2	lo
fe80::/128		::	U	0	0	2	lo
fe80::/128		::	U	0	0	2	lo
fe80::/128		::	U	0	0	2	lo
fe80::20e:2eff:fe6c:c0b5/128		::	U	0	34	2	lo
fe80::20e:2eff:fe6c:c0b5/128		::	U	0	0	2	lo
fe80::2e0:18ff:fe43:687/128		::	U	0	16	2	lo

▪ mobilní uzel zapojen v navštívené síti – komunikace s korespondentem

Kernel IPv6 routing table		Next Hop	Flags	Metric	Ref	Use	Iface
Destination							
2001:660:5503:276b:2/128		2001:660:5503:276b::2	UC	0	1	0	ip6tnl1
2001:660:5503:276b:2/128		2001:660:5503:276b::2	UC	0	1	1	eth1
2001:660:5503:276b:2/128		::	U	128	0	0	eth1
2001:660:5503:276b:2/128		::	U	192	0	0	ip6tnl1
2001:660:5503:276a:217:8ff:fe30:2d4/128		2001:660:5503:276a:217:8ff:fe30:2d4	UC	0	117	3	eth0
2001:660:5503:276a:2e0:4cff:fedb:1a16/128		2001:660:5503:276a:2e0:4cff:fedb:1a16	UC	0	1	0	eth0
2001:660:5503:276a::/64		::	U	256	1	0	eth0
2001:660:5503:276b::/64		::	U	256	0	0	eth1
fe80::/64		::	U	256	0	0	eth0
fe80::/64		::	U	256	0	0	eth1
fe80::/64		::	U	256	0	0	ip6tnl1
ff02::1/128		ff02::1	UC	0	158	0	eth0
ff02::1/128		ff02::1	UC	0	159	1	eth1
ff00::/8		::	U	256	0	0	eth0
ff00::/8		::	U	256	0	0	eth1
ff00::/8		::	U	256	0	0	ip6tnl1
::1/128		::	U	0	0	2	lo
2001:660:5503:276a::/128		::	U	0	0	2	lo
2001:660:5503:276a::1/128		::	U	0	53	2	lo
2001:660:5503:276b::/128		::	U	0	0	2	lo
2001:660:5503:276b::1/128		::	U	0	207	2	lo
2001:660:5503:276b:fdff:ffff:ffff:fffe/128		::	U	0	0	2	lo
fe80::/128		::	U	0	0	2	lo
fe80::/128		::	U	0	0	2	lo
fe80::/128		::	U	0	0	2	lo
fe80::20e:2eff:fe6c:c0b5/128		::	U	0	137	2	lo
fe80::20e:2eff:fe6c:c0b5/128		::	U	0	0	2	lo
fe80::2e0:18ff:fe43:687/128		::	U	0	32	2	lo

Mobilní uzel

▪ mobilní uzel zapojen v domácí síti

Kernel IPv6 routing table			
Destination	Next Hop	Flags Metric Ref	Use Iface
2001:660:5503:276b::/64	::	UA 256 632	0 eth0
fe80::/64	::	U 256 0	0 ip6tnl1
fe80::/64	::	U 256 0	0 eth0
ff02::1/128	ff02::1	UC 0 1	1 eth0
ff02::1/128	ff02::1	UC 0 644	0 eth0
ff00::/8	::	U 256 0	0 ip6tnl1
ff00::/8	::	U 256 0	0 eth0
::/0	fe80::2e0:18ff:fe43:687	UGDA 1024 1	0 eth0
::1/128	::	U 0 0	2 lo
2001:660:5503:276b::2/128	::	U 0 1	2 lo
2001:660:5503:276b:217:8ff:fe30:2d4/128	::	U 0 0	2 lo
fe80::217:8ff:fe30:2d4/128	::	U 0 0	2 lo
fe80::217:8ff:fe30:2d4/128	::	U 0 0	2 lo

▪ mobilní uzel zapojen v navštívené síti

root@0[~]# route -A inet6			
Kernel IPv6 routing table			
Destination	Next Hop	Flags Metric Ref	Use Iface
::/0	::	U 128 0	0 ip6tnl1
2001:660:5503:276b::1/128	2001:660:5503:276b::1	UC 0 2	2 ip6tnl1
2001:660:5503:276a::/64	::	UA 256 34	0 eth0
fe80::/64	::	U 256 0	0 ip6tnl1
fe80::/64	::	U 256 0	0 eth0
ff02::1/128	ff02::1	UC 0 35	0 eth0
ff00::/8	::	U 256 0	0 ip6tnl1
ff00::/8	::	U 256 0	0 eth0
::/0	fe80::20e:2eff:fe6c:c0b5	UGDA 1024 4	2 eth0
::1/128	::	U 0 0	2 lo
2001:660:5503:276a:217:8ff:fe30:2d4/128	::	U 0 3	2 lo
2001:660:5503:276b::2/128	::	U 0 2	2 lo
fe80::217:8ff:fe30:2d4/128	::	U 0 0	2 lo
fe80::217:8ff:fe30:2d4/128	::	U 0 0	2 lo

▪ mobilní uzel zapojen v navštívené síti – komunikace s korespondentem

Kernel IPv6 routing table			
Destination	Next Hop	Flags Metric Ref	Use Iface
::/0	::	U 128 0	0 ip6tnl1
2001:660:5503:276a:2e0:4cff:fedb:1a16/128	2001:660:5503:276a:2e0:4cff:fedb:1a16	UC 0 84	42 ip6tnl1
2001:660:5503:276b::1/128	2001:660:5503:276b::1	UC 0 2	1 ip6tnl1
2001:660:5503:276a:2e0:4cff:fedb:1a16/128	2001:660:5503:276a:2e0:4cff:fedb:1a16	UAC 0 43	42 eth0
2001:660:5503:276a::/64	::	UA 256 43	0 eth0
fe80::/64	::	U 256 0	0 ip6tnl1
fe80::/64	::	U 256 0	0 eth0
ff02::1/128	ff02::1	UC 0 44	0 eth0
ff00::/8	::	U 256 0	0 ip6tnl1
ff00::/8	::	U 256 0	0 eth0
::/0	fe80::20e:2eff:fe6c:c0b5	UGDA 1024 7	2 eth0
::1/128	::	U 0 0	2 lo
2001:660:5503:276a:217:8ff:fe30:2d4/128	::	U 0 89	2 lo
2001:660:5503:276b::2/128	::	U 0 86	2 lo
fe80::217:8ff:fe30:2d4/128	::	U 0 0	2 lo
fe80::217:8ff:fe30:2d4/128	::	U 0 4	2 lo

Korespondent

▪ korespondent zapojen v navštívené síti

Kernel IPv6 routing table			
Destination	Next Hop	Flags Metric Ref	Use Iface
2001:660:5503:276a::/64	::	UA 256 141	0 eth0
fe80::/64	::	U 256 0	0 eth0
ff02::1/128	ff02::1	UC 0 142	0 eth0
ff00::/8	::	U 256 0	0 eth0
::/0	fe80::20e:2eff:fe6c:c0b5	UGDA 1024 6	0 eth0
::1/128	::	U 0 0	2 lo
2001:660:5503:276a::3/128	::	U 0 0	2 lo
2001:660:5503:276a:2e0:4cff:fedb:1a16/128	::	U 0 4	2 lo
fe80::2e0:4cff:fedb:1a16/128	::	U 0 0	2 lo

▪ korespondent zapojen v navštívené síti – komunikace s mobilním uzlem

Kernel IPv6 routing table			
Destination	Next Hop	Flags Metric Ref	Use Iface
2001:660:5503:276a:217:8ff:fe30:2d4/128	2001:660:5503:276a:217:8ff:fe30:2d4	UAC 0 24	21 eth0
2001:660:5503:276a::/64	::	UA 256 1150	0 eth0
fe80::/64	::	U 256 0	0 eth0
ff02::1/128	ff02::1	UC 0 1151	0 eth0
ff00::/8	::	U 256 0	0 eth0
::/0	fe80::20e:2eff:fe6c:c0b5	UGDA 1024 707	22 eth0
::1/128	::	U 0 0	2 lo
2001:660:5503:276a::3/128	::	U 0 250	2 lo
2001:660:5503:276a:2e0:4cff:fedb:1a16/128	::	U 0 438	2 lo
fe80::2e0:4cff:fedb:1a16/128	::	U 0 49	2 lo

E. Výpisy síťové komunikace z aplikace Ethereal (příloha na CD)

Vzhledem k velkému rozsahu výpisů (26 stran) je tato příloha umístěna na příloženém CD v adresáři */Příloha E - Výpisy aplikace Ethereal/*.